```
<BlockBehavior>
	<General>
		<BasicValues>
			<DefensiveEffectMaxPercentMassMult>10</DefensiveEffectMaxPercentMassMult>	<!-- multiplication of mass used as max percent for defensive effects: ratio = effectBlockCount / (mass*DefensiveEffectMaxPercentMassMult)-->
			<ShieldExtraCapacityMultPerUnit>1</ShieldExtraCapacityMultPerUnit>	<!-- totalUnitShieldRecharge = (unitSize*ShieldExtraRechargeMultPerUnit) -->
			<ShieldExtraRechargeMultPerUnit>1</ShieldExtraRechargeMultPerUnit>	<!-- shieldCapacity = ((totalUnitShieldCapacity*ShieldCapacityPreMul)^ShieldCapacityPow)*ShieldCapacityTotalMul -->
			<ShieldDoInitialWithoutFromCore>true</ShieldDoInitialWithoutFromCore> <!-- if true, the ship will have initial shields without any shield blocks. if false, the initial values are added when the first shield block is placed -->
			<ShieldCapacityInitial>220</ShieldCapacityInitial> <!-- initial shield capacity on ships without shieldCapacity blocks -->
			<ShieldRechargeInitial>0</ShieldRechargeInitial> <!-- initial shield recharge on ships without shieldRecharge blocks -->
			<ShieldCapacityPow>0.9791797578</ShieldCapacityPow>
			<ShieldCapacityPreMul>1.0</ShieldCapacityPreMul>
			<ShieldCapacityTotalMul>110</ShieldCapacityTotalMul>	<!-- shieldCapacity = ((totalUnitShieldCapacity*ShieldCapacityPreMul)^ShieldCapacityPow)*ShieldCapacityTotalMul -->
			<ShieldRechargePow>1.0</ShieldRechargePow>	<!-- shieldCapacity = ((totalUnitShieldCapacity*1.0)^0.9)*55 -->
			<ShieldRechargePreMul>1.0</ShieldRechargePreMul>
			<ShieldRechargeTotalMul>5.5</ShieldRechargeTotalMul>
			<ShieldRechargeCycleTime>1.0</ShieldRechargeCycleTime> <!-- time in secs between recharge cycles. This value is relative, so changing it won't affect the time a ship needs to fully recharge (e.g. twice as many cycles -> half as many reacharge per cycle)-->
			<ShieldRecoveryTimeAfterOutage>10</ShieldRecoveryTimeAfterOutage> <!-- time, the shield will not recharge after reaching 0 -->
			<ShieldDirectRecoveryTime>60</ShieldDirectRecoveryTime> <!-- this time is set to the value gives after each hit sustained. In direct recovery, <ShieldRecoveryMultPerPercent> is applied. Otherwise it's running on full recharge -->
			<ShieldRecoveryMultPerPercent>0.5</ShieldRecoveryMultPerPercent> <!-- MAX % at high health	- in direct recovery, the shield recharge will be multiplied by (1-((shield/max)*thisValue));-->
			<ShieldRecoveryMult>0.25</ShieldRecoveryMult> <!-- MAX % at low health	- in direct recovery, the shield recharge will be multiplied: recharge = recharge*thisValue;-->
			<ShieldRechargePowerConsuption>10</ShieldRechargePowerConsuption> <!-- (when shield < 100%) per regen point -->
			<ShieldFullPowerConsuption>1</ShieldFullPowerConsuption> <!-- (when shield == 100%) per regen point -->
			<ShieldDockTransferLimit>0.25</ShieldDockTransferLimit> <!-- Up to which fill status a rail docked entity up in the chain (towards root) will take the hit (e.g. 0.25 means that a mothership (and docks inbetween) will take the shots of a turret or other rail dock as long as they have more than 25% of their max shields left -->
			<PowerDivFactor>0.333</PowerDivFactor>
			<PowerCeiling>2000000.0</PowerCeiling>
			<PowerGrowth>1.000348</PowerGrowth>
```

<PowerLinearGrowth>25.0</PowerLinearGrowth>
<PowerRecoveryTime>1000</PowerRecoveryTime>
<PowerBaseCapacity>50000</PowerBaseCapacity>
<PowerTankCapacityLinear>1000</PowerTankCapacityLinear> <!-- multiplication per unit -->
<PowerTankCapacityPow>1.05</PowerTankCapacityPow> <!-- unitSize^x per unit -->

<PowerBatteryGroupMultiplier>0.00613</PowerBatteryGroupMultiplier> <!-- recharge function: [recharge = (groupSize * <PowerBatteryGroupMultiplier>)^<PowerBatteryGroupPow>] -->
<PowerBatteryGroupPow>2.25</PowerBatteryGroupPow>

<PowerBatteryGroupCeiling>2000000.0</PowerBatteryGroupCeiling>
<PowerBatteryGroupGrowth>1.000232</PowerBatteryGroupGrowth>

<PowerBatteryLinearGrowth>25.0</PowerBatteryLinearGrowth>
<PowerBatteryCapacityLinear>500</PowerBatteryCapacityLinear> <!-- multiplication per unit -->
<PowerBatteryCapacityPow>1.10</PowerBatteryCapacityPow> <!-- unitSize^x per unit -->
<PowerBatteryTransferPercentRatePerSec>0.25</PowerBatteryTransferPercentRatePerSec> <!-- percent of max battery power transferred per second when active -->

<PowerBatteryTurnedOnRegenMultiplier>1.0</PowerBatteryTurnedOnRegenMultiplier> <!-- battery power regenration multipliler when turned on -->
<PowerBatteryTurnedOffRegenMultiplier>0.2</PowerBatteryTurnedOffRegenMultiplier> <!-- battery power regenration multipliler when turned off -->
<PowerBatteryTransferTopOffOnly>false</PowerBatteryTransferTopOffOnly> <!-- if true, an active battery will only top off the normal power of an entity, and not unload any further energy if the normal power is at 100%-->

<PowerBatteryExplosionCountMaxPercent>0.005</PowerBatteryExplosionCountMaxPercent> <!-- max explosions per group, % of group size-->

<PowerBatteryExplosionCountPerBlocksInGroup>0.1</PowerBatteryExplosionCountPerBlocksInGroup> <!-- explosions per group, % of group size -->
<PowerBatteryExplosionCountMax>100</PowerBatteryExplosionCountMax> <!-- max explosions per group -->

<PowerBatteryExplosionsPerSecond>0.5</PowerBatteryExplosionsPerSecond> <!-- explosions per second -->

<PowerBatteryExplosionRadiusPerBlocksInGroup>0.001</PowerBatteryExplosionRadiusPerBlocksInGroup>> <!-- radius of a explosion based on the group's member count, minimum 1 -->
<PowerBatteryExplosionRadiusMax>8.0</PowerBatteryExplosionRadiusMax> <!-- capped at 64 -->

<PowerBatteryExplosionDamagePerBlocksInGroup>2.0</PowerBatteryExplosionDamagePerBlocksInGroup> <!-- damage of a explosion based on the group's member count -->
<PowerBatteryExplosionDamageMax>100000.0</PowerBatteryExplosionDamageMax> <!-- capped at max signed integer -->

<TakeOffEffectPowerConsumptionMult>1</TakeOffEffectPowerConsumptionMult> <!-- multiplied by force -->
<EvadeEffectPowerConsumptionMult>1</EvadeEffectPowerConsumptionMult> <!-- multiplied by force -->
<PersonalSalvageBeamBonus>1.25</PersonalSalvageBeamBonus> <!-- bonus mult of raw resources when salvaging with handheld salvage beam (everything between whole numbers is chance (e.g. 1.25 give 25% chance of an extra raw yield) -->

<RailMassEnhancerFreeMass>50.0</RailMassEnhancerFreeMass> <!-- mass that is free to place without the need of a rail mass enhancer -->
<RailMassEnhancerMassPerEnhancer>5.0</RailMassEnhancerMassPerEnhancer> <!-- how much mass is free to place for each rail mass enhancer in the structure -->

<RailMassEnhancerPowerConsumedPerEnhancer>50.0</RailMassEnhancerPowerConsumedPerEnhancer> <!-- How much power per second is consumed my rail enhancer blocks -->

<RailMassEnhancerPercentCostPerMassAboveEnhancerProvided>0.05</RailMassEnhancerPercentCostPerMassAboveEnhancerProvided> <!-- how much % of mass speed (between 0 and 1) is subtracted for every one mass placed on a dock that is above the current mass limit (free blocks + enhancers) (e.g. at 0.1, for every block placed the speed is one percent slower)-->

<PlanetPowerBaseCapacity>500</PlanetPowerBaseCapacity>
<AsteroidPowerBaseCapacity>500</AsteroidPowerBaseCapacity>

<StructureHpBlockMultiplier>1.0</StructureHpBlockMultiplier> <!-- how much of the hp value is added to the systemHP -->
<HpDeductionLogOffset>-5</HpDeductionLogOffset>
<HpDeductionLogFactor>0.7</HpDeductionLogFactor> <!-- deducted hit points on damage: blockHP + HpDeductionLogFactor*(max(0, log10(maxHP)+HpDeductionLogOffset)) * blockHP -->

<ShipRebootTimeInSecPerMissingHpPercent>40.0</ShipRebootTimeInSecPerMissingHpPercent> <!-- on 0% hp, the full value in seconds is needed to reboot -->
<ShipRebootTimeMultiplierPerMass>0.0002</ShipRebootTimeMultiplierPerMass> <!-- secondsToReboot * (mass * ShipRebootTimeMultiplyerPerMass) -->
<ShipRebootTimeMinSec>30.0</ShipRebootTimeMinSec> <!-- Minimum time in secs a reboot will take -->

<ArmorHpAbsorbtion>0.5</ArmorHpAbsorbtion> <!-- absorbedDamageToArmorHP = ArmorHpAbsorbtion; How much damage of a shot can be absorbed to armorHP; this is a static value; set it to 0 if you want to use a dynamic variance instead with ArmorHpDamageThreshold-->

<ArmorHpAbsorbedDamageMultiplier>1.0</ArmorHpAbsorbedDamageMultiplier> <!-- How much absorbed damaged is deducted from armorHP -->

<ArmorHpBlockMultiplier>1.0</ArmorHpBlockMultiplier> <!-- how much of the armor value is added to the armorHP -->

<ArmorHpDamageThreshold>1.0</ArmorHpDamageThreshold> <!-- absorbedDamageToArmorHP = (ArmorHpDamageThreshold * [armorhp of the block]) * [current ArmorHp Left On Ship]; Note: this is not used as long as ArmorHpAbsorbtion != 0; This means, a ship wont take damage other than armorHP damage as long as a single shot isn't above the trheshold; -->

<AITurretMinOrientationSpeed>0.5</AITurretMinOrientationSpeed> <!-- Minimum turret orientation speed by mass used by AI (still depends on enhancer) -->

<AITurretMaxOrientationSpeed>3.5</AITurretMaxOrientationSpeed> <!-- Maximum turret orientation speed by mass used by AI (still depends on enhancer) -->

<AITurretOrientationSpeedDivByMass>50.0</AITurretOrientationSpeedDivByMass> <!-- speed = AITurretOrientationSpeedDivByMass / mass -->

<ExplosionShieldDamageBonus>0.0</ExplosionShieldDamageBonus> <!-- all explosions (missile/warhead/pulse) do [damage + damage * bonus] against shields-->

<ExplosionHullDamageBonus>0.0</ExplosionHullDamageBonus> <!-- all explosions (missile/warhead/pulse) do [damage + damage * bonus] against blocks (will also hurt armor more)-->

<VolumeMassMultiplier>0.01</VolumeMassMultiplier> <!-- how much of the volume of an item in an inventory/storage/cargo adds to the mass of a structure -->

<HpConditionTriggerList>

<Item conditionhp="0.9">
<Type>POWER</Type>
<Amount>0.9</Amount>
</Item>
<Item conditionhp="0.9">
<Type>SHIELD</Type>
<Amount>0.9</Amount>
</Item>
<Item conditionhp="0.9">
<Type>THRUST</Type>
<Amount>0.9</Amount>
</Item>

<Item conditionhp="0.8">
<Type>POWER</Type>
<Amount>0.8</Amount>
</Item>
<Item conditionhp="0.8">
<Type>SHIELD</Type>
<Amount>0.8</Amount>
</Item>
<Item conditionhp="0.8">
<Type>THRUST</Type>
<Amount>0.8</Amount>
</Item>

<Item conditionhp="0.7">
<Type>POWER</Type>
<Amount>0.7</Amount>
</Item>
<Item conditionhp="0.7">
<Type>SHIELD</Type>
<Amount>0.7</Amount>
</Item>
<Item conditionhp="0.7">
<Type>THRUST</Type>
<Amount>0.7</Amount>
</Item>

<Item conditionhp="0.6">
<Type>POWER</Type>
<Amount>0.5</Amount>
</Item>
<Item conditionhp="0.6">
<Type>SHIELD</Type>
<Amount>0.5</Amount>
</Item>
<Item conditionhp="0.6">
<Type>THRUST</Type>
<Amount>0.5</Amount>
</Item>

<Item conditionhp="0.55">
<Type>CONTROL_LOSS</Type>
</Item>

<Item conditionhp="0.5">
<Type>OVERHEATING</Type>

```
                </Item>
                                        <Item conditionhp="0.5">
                  <Type>POWER</Type>
                  <Amount>0.0</Amount>
                </Item>
                <Item conditionhp="0.5">
                  <Type>SHIELD</Type>
                  <Amount>0.0</Amount>
                </Item>
                <Item conditionhp="0.5">
                  <Type>THRUST</Type>
                  <Amount>0.5</Amount>
                </Item>
              </HpConditionTriggerList>

        </BasicValues>
    </General>
    <Thruster>
        <BasicValues>
            <MulTotalThrust>0.75</MulTotalThrust> <!--  total = (total^PowTotalThrust)*MulTotalThrust -->
            <PowTotalThrust>0.87</PowTotalThrust> <!--  total = (total^PowTotalThrust)*MulTotalThrust -->
            <ThrustPowerconsumptionPerBlock>1</ThrustPowerconsumptionPerBlock> <!-- WARNING: this is
consumed every 30ms! use -1 for power consuption per thrust generated -->
            <PowPerThrustCollection>1</PowPerThrustCollection>
            <UnitCalcStyle>LINEAR</UnitCalcStyle> <!-- LINEAR, BOX_DIM_MULT, BOX_DIM_ADD -->
            <UnitCalcMult>5.5</UnitCalcMult> <!-- multiplied with result of UnitCalcStyle -->

            <MinThrustMassRatio>0.0</MinThrustMassRatio> <!-- minimum thrust to mass ratio multiplied with
server max speed -->
            <MaxThrustMassRatio>2.5</MaxThrustMassRatio> <!-- maximum thrust to mass ratio multiplied with
server max speed -->
            <ThrustMassRatioMaxSpeedAddition>0.5</ThrustMassRatioMaxSpeedAddition> <!-- additional add to
ratio -->
            <ThrustMassRatioMaxSpeedMultiplier>1.0</ThrustMassRatioMaxSpeedMultiplier> <!-- additional
multiplier of ratio -->

            <ThrustRotPercentMult>50.0</ThrustRotPercentMult> <!-- additional multiplier of ratio -->
            <InertiaPow>0.30</InertiaPow> <!-- additional multiplier of ratio -->

            <ThrustBalanceChangeApplyTimeInSecs>15.0</ThrustBalanceChangeApplyTimeInSecs> <!-- how
much time it takes for a thrust balance change to take effect-->

        </BasicValues>
    </Thruster>
    <Cloaking>
        <BasicValues>
```

```
            <PowerConsumedPerSecondPerBlock>145</PowerConsumedPerSecondPerBlock>

            <ReuseDelayOnActionMs>6000</ReuseDelayOnActionMs> <!-- use -1 here to not switch off on action
(e.g. shooting) -->
            <ReuseDelayOnModificationMs>6000</ReuseDelayOnModificationMs>
            <ReuseDelayOnSwitchedOffMs>1000</ReuseDelayOnSwitchedOffMs>
            <ReuseDelayOnHitMs>10000</ReuseDelayOnHitMs>
            <ReuseDelayOnScanMs>30000</ReuseDelayOnScanMs>
            <ReuseDelayOnNoPowerMs>6000</ReuseDelayOnNoPowerMs>

        </BasicValues>
    </Cloaking>
    <Jamming>
        <BasicValues>
            <PowerConsumedPerSecondPerBlock>5</PowerConsumedPerSecondPerBlock>

             <ReuseDelayOnActionMs>-1</ReuseDelayOnActionMs> <!-- use -1 here to not switch off on action
(e.g. shooting) -->
            <ReuseDelayOnModificationMs>6000</ReuseDelayOnModificationMs>
            <ReuseDelayOnSwitchedOffMs>1000</ReuseDelayOnSwitchedOffMs>
            <ReuseDelayOnHitMs>10000</ReuseDelayOnHitMs>
            <ReuseDelayOnScanMs>30000</ReuseDelayOnScanMs>
            <ReuseDelayOnNoPowerMs>6000</ReuseDelayOnNoPowerMs>
        </BasicValues>
    </Jamming>

    <JumpDrive>
        <BasicValues>
                <RatioNeededToTotalBlockCount>0.05</RatioNeededToTotalBlockCount>
                        <ChargeNeededForJump>230000</ChargeNeededForJump>
                        <ChargeNeededForJumpPerBlock>1200</ChargeNeededForJumpPerBlock> <!--
(in drive module blocks) added to <chargeNeededForJump> -->

<ChargeNeededForJumpPerBlockAfterRatioMet>1</ChargeNeededForJumpPerBlockAfterRatioMet> <!-- (in drive
module blocks for every block after ratio is met -->
                        <ReloadMsAfterUseMs>10</ReloadMsAfterUseMs>
                        <ChargeAddedPerSecond>10000</ChargeAddedPerSecond>
                        <ChargeAddedPerSecondPerBlock>50</ChargeAddedPerSecondPerBlock> <!--
(in drive module blocks) -->
                        <DistanceInSectors>8</DistanceInSectors>
        </BasicValues>
    </JumpDrive>

    <JumpInhibitor>
        <BasicValues>
                        <DischargePerSecond>10000</DischargePerSecond>
```

```
                                        <DischargePerSecondPerBlock>50</DischargePerSecondPerBlock> <!-- (in
module blocks) -->
                                        <PowerConsumptionPerSecond>20000</PowerConsumptionPerSecond>

<PowerConsumptionPerSecondPerBlock>100</PowerConsumptionPerSecondPerBlock> <!-- (in module blocks) -->
                                        <DischargeFriendlyShips>true</DischargeFriendlyShips>
                                        <DischargeSelf>true</DischargeSelf>
            </BasicValues>
        </JumpInhibitor>

        <Scanner>
            <BasicValues>
                        <RatioNeededToTotalBlockCount>0.05</RatioNeededToTotalBlockCount>
                                        <ChargeNeededForScan>110000</ChargeNeededForScan>
                                        <ChargeNeededForScanPerBlock>1200</ChargeNeededForScanPerBlock> <!--
(in drive module blocks) added to <chargeNeededForScan> -->

<ChargeNeededForScanPerBlockAfterRatioMet>1</ChargeNeededForScanPerBlockAfterRatioMet> <!-- (in drive
module blocks for every block after ratio is met -->
                                        <ReloadMsAfterUseMs>10</ReloadMsAfterUseMs>
                                        <DefaultScanDistance>4</DefaultScanDistance>
                                        <EnemySystemDistanceMult>0.5</EnemySystemDistanceMult> <!-- Scan
distance modifier for scanners in an enemy territory -->
                                        <AllySystemDistanceMult>1.5</AllySystemDistanceMult> <!-- Scan distance
modifier for allied and own territory (owner additionally scans whole 16x16x16 sector system)-->
                                        <ChargeAddedPerSecond>5000</ChargeAddedPerSecond>
                                        <ChargeAddedPerSecondPerBlock>25</ChargeAddedPerSecondPerBlock> <!--
(in drive module blocks) -->
            </BasicValues>
        </Scanner>
        <ShipYard>
            <BasicValues>
                <PowerNeededPerShipyardBlock>50</PowerNeededPerShipyardBlock> <!-- constant power
consumption to keep the ship yard active per second -->
                <ShipyardArcMaxSpacing>50</ShipyardArcMaxSpacing> <!-- max distance between shipyard arcs -->
                <ConstructionTickInSeconds>1</ConstructionTickInSeconds> <!-- construction tick in seconds -->
                <ConstructionBlocksTakenPerTick>2500</ConstructionBlocksTakenPerTick> <!-- blocks constructed per
tick -->
                <DeconstructionTimePerBlockInMilliseconds>0.01</DeconstructionTimePerBlockInMilliseconds> <!--
time in milliseconds added to constant time per block -->
                <DeconstructionConstantTimeInMilliseconds>10000</DeconstructionConstantTimeInMilliseconds> <!--
constant time to deconstruct a ship-->
            </BasicValues>
        </ShipYard>
        <WarpGate>
            <BasicValues>
```

```
            <PowerNeededPerGateBlock>100</PowerNeededPerGateBlock> <!-- constant power consumption to keep the gate active -->
            <PowerNeededPerMass>100</PowerNeededPerMass> <!-- to Jump, the One Time Power Expense is shipMass * PowerNeededPerMass at the time of jumping-->
            <DistanceInSectors>128</DistanceInSectors> <!-- default: 8 systems -->
        </BasicValues>
    </WarpGate>
    <RaceGate>
        <BasicValues>
            <PowerNeededPerGateBlock>0</PowerNeededPerGateBlock> <!-- constant power consumption to keep the gate active -->
            <PowerNeededPerMass>0</PowerNeededPerMass> <!-- to Jump, the One Time Power Expense is shipMass * PowerNeededPerMass at the time of jumping-->
        </BasicValues>
    </RaceGate>
    <ActivationGate>
        <BasicValues>
            <PowerNeededPerGateBlock>0</PowerNeededPerGateBlock> <!-- constant power consumption to keep the gate active -->
            <PowerNeededPerMass>0</PowerNeededPerMass> <!-- to Jump, the One Time Power Expense is shipMass * PowerNeededPerMass at the time of jumping-->
        </BasicValues>
    </ActivationGate>
    <Transporter>
        <BasicValues>
        <PowerNeededPerGateBlock>0</PowerNeededPerGateBlock> <!-- constant power consumption to keep the gate active -->
        </BasicValues>
    </Transporter>
    <Shop>
        <BasicValues>
            <TradingGuildCargoPerShip>1000.0</TradingGuildCargoPerShip>
            <TradingGuildCostPerSystem>300</TradingGuildCostPerSystem>
            <TradingGuildCostPerCargoShip>1000</TradingGuildCostPerCargoShip>
            <TradingGuildTimePerSectorSec>3.0</TradingGuildTimePerSectorSec>
            <TradingGuildProfitOfValue>0.05</TradingGuildProfitOfValue>
            <TradingGuildProfitOfValuePerSystem>0.01</TradingGuildProfitOfValuePerSystem>

        </BasicValues>
    </Shop>
    <Explosive>
        <BasicValues>
            <BlockDamage>2000</BlockDamage>
            <PlayerDamage>100</PlayerDamage>
            <Radius>8</Radius>
        </BasicValues>
```

```
    </Explosive>
    <Cargo>
                <BasicValues>
                    <CapacityPerBlockMultShip>100.0</CapacityPerBlockMultShip>  <!-- simple capacity per block added -->
                    <CapacityPerBlockMultStation>100.0</CapacityPerBlockMultStation>  <!-- simple capacity per block added -->
            <CapacityPerGroupQuadraticShip>1.0</CapacityPerGroupQuadraticShip> <!-- determines group behavior. calculated: (groupsize*CapacityPerBlocMult) ^ CapacityPerGroupQuadratic  -->
            <CapacityPerGroupQuadraticStation>1.0</CapacityPerGroupQuadraticStation> <!-- determines group behavior. calculated: (groupsize*CapacityPerBlocMult) ^ CapacityPerGroupQuadratic  -->

                    <InventoryBaseCapacityShip>100</InventoryBaseCapacityShip> <!-- base capacity for all block based inventories -->
                    <InventoryBaseCapacityStation>100</InventoryBaseCapacityStation> <!-- base capacity for all block based inventories -->

                    <PercentageBledPerMinute>0.1</PercentageBledPerMinute> <!-- how much inventory is lost per minute if over capacity (must be between [0, 1]) -->

                    <PersonalInventoryBaseCapacity>2000</PersonalInventoryBaseCapacity> <!-- base capacity for the personal player inventory-->
                    <PersonalFactoryBaseCapacity>100</PersonalFactoryBaseCapacity> <!-- base inventory for the personal player factories -->
                </BasicValues>
    </Cargo>
    <SalvageBeam>
        <BasicValues>
            <TickRate>0.025</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer resets -->
            <SalvageDamageNeededPerBlock>200</SalvageDamageNeededPerBlock> <!-- 0 will salvage on tick, otherwise the damage has to be met on the object to salvage one block -->
                                <SalvageDamagePerHit>2</SalvageDamagePerHit> <!-- salvage damage done per tick to fill up SalvageDamageNeededPerBlock -->
            <PowerConsumptionPerTick>0.4</PowerConsumptionPerTick>
            <Distance>0.12</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 / (unitSize^pow)*mult -->
            <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
            <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
            <InitialTicks>10</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
            <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Salvage Damage percent done to target in own Rail Chain when target is equal or closer to root than this entity -->
            <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Salvage Damage percent done to target in own Rail Chain when target is further from root than this entity -->

        </BasicValues>
```

```
        <Combination>
            <Cannon>
                <HitSpeed style="skip" />
                <Distance style="skip" />
                <PowerConsumption style="skip" linear="true" />
                <PowerPerHit style="nerf" value="2" linear="true" />
                <Split style="skip" />
                <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Cannon>
            <Missile>
                <HitSpeed style="skip" />
                <Distance style="skip" />
                <PowerConsumption style="skip" linear="true"/>
                <PowerPerHit style="skip" linear="true"/>
                <Split style="buff" value="9"/>
                <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated
-->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Missile>
            <Beam>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
                <Distance style="buff" value="2" />
                <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                <PowerPerHit style="buff" linear="true" value="2" />
                <Split style="skip" />
                <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Beam>
            <Pulse>
                <HitSpeed style="nerf" inverse="true" value="5" />
                <Distance style="skip" />
                <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                <PowerPerHit style="buff" value="5" linear="true" />
                <Split style="skip" />
                <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Pulse>
        </Combination>
```

```
    </SalvageBeam>
    <PowerDrain>
        <BasicValues>
            <TickRate>0.5</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer resets -->
            <DrainPerHit>100</DrainPerHit>
            <PowerConsumptionOnTargetFromBeamPower>1.0</PowerConsumptionOnTargetFromBeamPower> <!-- percentage (0 to 1) of SupplyPerHit that consumed on target -->
            <PowerIncOnShooterFromBeamPower>0.75</PowerIncOnShooterFromBeamPower> <!-- percentage (0 to 1) of drain that gets added to shooter -->
            <PowerConsumptionPerTick>25</PowerConsumptionPerTick>
            <Distance>0.24</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 / (unitSize^pow)*mult -->
            <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
            <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
            <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
            <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Drain percent done to target in own Rail Chain when target is equal or closer to root than this entity -->
            <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Drain percent done to target in own Rail Chain when target is further from root than this entity -->
        </BasicValues>
        <Combination>
            <Cannon>
                <HitSpeed style="buff" inverse="true" value="10" />
                <Distance style="skip" />
                <PowerConsumption style="buff" inverse="true" value="20" linear="true" />
                <PowerPerHit style="nerf" value="20" linear="true" />
                <Split style="skip" />
                <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Cannon>
            <Missile>
                <HitSpeed style="skip" />
                <Distance style="skip" />
                <PowerConsumption style="skip" linear="true"/>
                <PowerPerHit style="skip" linear="true" />
                <Split style="buff" value="9"/>
                <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Missile>
            <Beam>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
```

```
                    <Distance style="buff" value="2" />
                    <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                    <PowerPerHit style="buff" linear="true" value="2" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Beam>
                <Pulse>
                    <HitSpeed style="nerf" inverse="true" value="5" />
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                    <PowerPerHit style="buff" value="5" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </PowerDrain>
        <PowerSupply>
            <BasicValues>
                <TickRate>0.5</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer
resets -->
                <PowerIncOnTargetFromBeamPower>1.0</PowerIncOnTargetFromBeamPower> <!-- percentage (0 to
1) of SupplyPerHit that gets added to the target -->

<PowerConsumptionOnShooterFromBeamPower>0.0</PowerConsumptionOnShooterFromBeamPower> <!--
percentage (0 to 1) of SupplyPerHit that consumed (additionally to PowerConsumptionPerTick) on shooter -->
                <SupplyPerHit>240</SupplyPerHit>
                <PowerConsumptionPerTick>300</PowerConsumptionPerTick>
                <Distance>0.24</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 /
(unitSize^pow)*mult -->
                <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
                <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
                <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Supply percent done to target in own Rail
Chain when target is equal or closer to root than this entity -->
                <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Supply percent done to target in own Rail Chain
when target is further from root than this entity -->
            </BasicValues>
            <Combination>
                <Cannon>
                    <HitSpeed style="buff" inverse="true" value="10" />
```

```
                    <Distance style="skip" />
                    <PowerConsumption style="buff" inverse="true" value="20" linear="true" />
                    <PowerPerHit style="nerf" value="20" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Cannon>
                <Missile>
                    <HitSpeed style="skip" />
                    <Distance style="skip" />
                    <PowerConsumption style="skip" linear="true"/>
                   <PowerPerHit style="skip" linear="true"/>
                    <Split style="buff" value="9"/>
                    <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated
-->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Missile>
                <Beam>
                    <HitSpeed style="skip" /> <!-- Affects TickRate -->
                    <Distance style="buff" value="2" />
                    <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                    <PowerPerHit style="buff" linear="true" value="2" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Beam>
                <Pulse>
                    <HitSpeed style="nerf" inverse="true" value="5" />
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                    <PowerPerHit style="buff" value="5" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </PowerSupply>
        <ShieldDrain>
            <BasicValues>
```

```
            <TickRate>0.5</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer resets -->
            <DrainPerHit>10</DrainPerHit>
            <PowerConsumptionPerTick>100</PowerConsumptionPerTick>
            <ShieldConsumptionOnTargetFromBeamPower>1.0</ShieldConsumptionOnTargetFromBeamPower> <!-- percentage (0 to 1) of beam power taken of target in shield -->
            <ShieldIncOnShooterFromBeamPower>0.5</ShieldIncOnShooterFromBeamPower> <!-- percentage (0 to 1) of beam power given to shooter in shield -->
            <Distance>0.24</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 / (unitSize^pow)*mult -->
            <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
            <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
            <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
            <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Drain percent done to target in own Rail Chain when target is equal or closer to root than this entity -->
            <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Drain percent done to target in own Rail Chain when target is further from root than this entity -->
        </BasicValues>
        <Combination>
            <Cannon>
                <HitSpeed style="buff" inverse="true" value="10" />
                <Distance style="skip" />
                <PowerConsumption style="buff" inverse="true" value="20" linear="true" />
                <PowerPerHit style="nerf" value="20" linear="true" />
                <Split style="skip" />
                <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Cannon>
            <Missile>
                <HitSpeed style="skip" />
                <Distance style="skip" />
                <PowerConsumption style="skip" linear="true"/>
                <PowerPerHit style="skip" linear="true"/>
                <Split style="buff" value="9"/>
                <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Missile>
            <Beam>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
                <Distance style="buff" value="2" />
                <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                <PowerPerHit style="buff" linear="true" value="2" />
```

```
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Beam>
                <Pulse>
                    <HitSpeed style="nerf" inverse="true" value="5" />
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                    <PowerPerHit style="buff" value="5" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </ShieldDrain>
        <ShieldSupply>
            <BasicValues>
                <TickRate>0.5</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer
resets -->
                <SupplyPerHit>240</SupplyPerHit>
                <PowerConsumptionPerTick>30</PowerConsumptionPerTick>

<ShieldConsumptionOnShooterFromBeamPower>1.0</ShieldConsumptionOnShooterFromBeamPower> <!--
percentage (0 to 1) of beam power taken from shooter in shield -->
                <ShieldIncOnTargetFromBeamPower>1.0</ShieldIncOnTargetFromBeamPower> <!-- percentage (0 to
1) of beam power given to shooter in shield -->
                <Distance>0.24</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 /
(unitSize^pow)*mult -->
                <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
                <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
                <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Supply percent done to target in own Rail
Chain when target is equal or closer to root than this entity -->
                <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Supply percent done to target in own Rail Chain
when target is further from root than this entity -->
            </BasicValues>
            <Combination>
                <Cannon>
                    <HitSpeed style="buff" inverse="true" value="10" />
                    <Distance style="skip" />
                    <PowerConsumption style="buff" inverse="true" value="20" linear="true" />
                    <PowerPerHit style="nerf" value="20" linear="true" />
```

```
                    <Split style="skip" />
                    <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Cannon>
                <Missile>
                    <HitSpeed style="skip" />
                    <Distance style="skip" />
                    <PowerConsumption style="skip" linear="true"/>
                    <PowerPerHit style="skip" linear="true"/>
                    <Split style="buff" value="9"/>
                    <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated
-->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Missile>
                <Beam>
                    <HitSpeed style="skip" /> <!-- Affects TickRate -->
                    <Distance style="buff" value="2" />
                    <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                    <PowerPerHit style="buff" linear="true" value="2" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Beam>
                <Pulse>
                    <HitSpeed style="nerf" inverse="true" value="5" />
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                    <PowerPerHit style="buff" value="5" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </ShieldSupply>
        <RepairBeam>
            <BasicValues>
                    <TickRate>0.5</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this
timer resets -->
                <RepairPerHit>60</RepairPerHit>
```

```
            <PowerConsumptionPerTick>240</PowerConsumptionPerTick>
            <Distance>0.12</Distance> <!-- 1 is equal to 100% sector radius --> <!-- timeBetweenHits = 1 /
(unitSize^pow)*mult -->
            <CoolDown>5</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
            <BurstTime>2.5</BurstTime> <!-- Time the beam will fire -->
            <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
            <RailHitMultiplierParent>1.0</RailHitMultiplierParent> <!-- Repair percent done to target in own Rail
Chain when target is equal or closer to root than this entity -->
            <RailHitMultiplierChild>1.0</RailHitMultiplierChild> <!-- Repair percent done to target in own Rail Chain
when target is further from root than this entity -->
        </BasicValues>
        <Combination>
            <Cannon>
                <HitSpeed style="buff" inverse="true" value="10" />
                <Distance style="skip" />
                <PowerConsumption style="buff" inverse="true" value="20" linear="true" />
                <PowerPerHit style="nerf" value="20" linear="true" />
                <Split style="skip" />
                <CoolDown style="buff" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Cannon>
            <Missile>
                <HitSpeed style="skip" />
                <Distance style="skip" />
                <PowerConsumption style="skip" linear="true"/>
                <PowerPerHit style="skip" linear="true"/>
                <Split style="buff" value="9"/>
                <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated
-->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Missile>
            <Beam>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
                <Distance style="buff" value="2" />
                <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                <PowerPerHit style="buff" linear="true" value="2" />
                <Split style="skip" />
                <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Beam>
            <Pulse>
```

```
                    <HitSpeed style="nerf" inverse="true" value="5" />
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                    <PowerPerHit style="buff" value="5" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                    <BurstTime style="buff" value="5"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </RepairBeam>
        <Trigger>
        </Trigger>
        <Activation>
        </Activation>
        <Sensor>
        </Sensor>
        <RailSpeed>
        </RailSpeed>
        <FafoMissile>       <!-- this module is going to be removed -->
        </FafoMissile>
        <HeatMissile>       <!-- this module is going to be removed -->
        </HeatMissile>
        <Factory>
        </Factory>
        <TurretDockingBlock>
        </TurretDockingBlock>
        <FixedDockingBlock>
        </FixedDockingBlock>
        <Cannon>
            <BasicValues>
                <Damage>10</Damage>
                <Distance>1</Distance> <!-- 1 is equal to 100% sector radius -->
                <Speed>12.5</Speed> <!-- In relation % to server max speed  1 = 100% -->
                <ReloadMs>1000</ReloadMs>
                <PowerConsumption>100</PowerConsumption>
                <AdditionalPowerConsumptionPerUnitMult>0.1</AdditionalPowerConsumptionPerUnitMult> <!-- Nerf based on amount of groups connected to the same controller: powerConsumption * (1+countGroups*thisValue); 0 to turn off nerf-->
                <DefaultWeaponPenetrationWeight>1.5</DefaultWeaponPenetrationWeight> <!-- This weight will distribute the damage on the penetration depth path. The higher, the more damage is used in the first blocks -->

<DefaultWeaponPenetrationDeepnessMultiplier>1.5</DefaultWeaponPenetrationDeepnessMultiplier> <!-- how deep is the penetration potentially (depending on log10 of damage). A value of 0 defaults to the old system (full damage hits one block) -->
```

```
        </BasicValues>
        <Combination>
            <Cannon>
                <Damage style="nerf" linear="true" value="10" />
                <Reload style="buff" inverse="true" value="10" />
                <Distance style="skip" />
                <Speed style="skip" />
                <Split style="skip" />
                <PowerConsumption style="nerf" linear="true" value="10" />
            </Cannon>
            <Missile>
                <Damage style="buff" linear="true" value="1" />
                <Reload style="nerf" inverse="true" value="1" />
                <Distance style="skip" />
                <Speed style="skip" />
                <Split style="buff" value="9" />
                <PowerConsumption style="nerf" inverse="true" linear="true" value="1"/>
            </Missile>
            <Beam>
                <Damage style="buff" linear="true" value="3" /><!-- 1+3 = 4x damage -->
                <Reload style="nerf" inverse="true" value="3" /><!-- 1+3 = 4000ms -->
                <Distance style="buff" value="2" />
                <Speed style="buff" value="1" />
                <Split style="skip" />
                <PowerConsumption linear="true" style="nerf" inverse="true" value="3"/> <!-- 1+3 = 4x power
consumption -->
            </Beam>
            <Pulse>
                <Damage style="buff" value="15" linear="true" />
                <Reload style="nerf" inverse="true" value="15" />
                <Distance style="skip" />
                <Speed style="skip" />
                <Split style="skip" />
                <PowerConsumption style="buff" linear="true" value="15" />
            </Pulse>
        </Combination>
    </Cannon>
    <DumbMissile>
        <BasicValues>
            <Damage>300</Damage>
            <HpPerDamage>0</HpPerDamage>
            <Distance>1.6</Distance> <!-- 1 is equal to 100% sector radius -->
            <Radius>12</Radius>                    <!--this a maximum blast radius -->
            <Speed>2.48</Speed>
            <ReloadMs>15000</ReloadMs>
```

```
            <PowerConsumption>1500</PowerConsumption> <!-- deviates from 10 power/1damage formula, missiles can easily be shot down -->
            <AdditionalPowerConsumptionPerUnitMult>0.025</AdditionalPowerConsumptionPerUnitMult> <!-- Nerf based on amount of groups connected to the same controller: powerConsumption * (1+countGroups*thisValue); 0 to turn off nerf-->
            <ChasingTurnSpeedWithTargetInFront>1.0</ChasingTurnSpeedWithTargetInFront>
            <ChasingTurnSpeedWithTargetInBack>0.5</ChasingTurnSpeedWithTargetInBack>


        </BasicValues>
        <Combination>
            <Cannon>
                <Damage style="nerf" value="15" linear="true" />
                <Reload style="buff" inverse="true" value="15" />
                <Distance style="skip" />
                <BlastRadius style="nerf" value="2" />
                <Speed style="buff" value="2.0"/>
                <Split style="skip" />
                <Mode style="set" value="0"/> <!-- Dumb Missile -->
                <PowerConsumption style="nerf" value="15" linear="true" />
            </Cannon>
            <Missile>
                <Damage style="skip" linear="true" />
                <Reload style="skip" />
                <Distance style="skip" />
                <BlastRadius style="nerf" value="3" />
                <Speed style="nerf" value="2" />
                <Split style="buff" value="9" />
                <Mode style="set" value="1"/><!-- Heat Seeker Missile -->
                <PowerConsumption style="skip" linear="true" />
            </Missile>
            <Beam>
                <Damage style="buff" linear="true" value="2" />
                <Reload style="nerf" inverse="true" value="2" />
                <Distance style="buff" value="2" />
                <BlastRadius style="skip" />
                <Speed style="buff" value="1" />
                <Split style="skip" />
                <Mode style="set" value="2" /><!-- Smart Missile -->
                <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
            </Beam>
            <Pulse>
                <Damage style="buff" linear="true" value="5"/>
                <Reload style="nerf" inverse="true" value="5" />
                <Distance style="skip" />
                <BlastRadius style="buff" value="3" />
```

```
                <Speed style="nerf" value="3" />
                <Split style="skip" />
                <Mode style="set" value="2" /><!-- Smart Missile -->
                <PowerConsumption style="nerf" inverse="true" linear="true" value="5"/>
            </Pulse>
        </Combination>
    </DumbMissile>
    <DamageBeam>
        <BasicValues>
            <DamagePerHit>10</DamagePerHit>
            <PowerConsumptionPerTick>100</PowerConsumptionPerTick>
            <Distance>0.5</Distance> <!-- 1 is equal to 100% sector radius, reduced range versus other weapons to compensate for beams instant travel time -->
            <TickRate>0.2</TickRate>  <!-- tick time in seconds, beam must remain on target entity or this timer resets -->
            <CoolDown>5.0</CoolDown> <!-- Time it takes to fire beam again from the start-time it first activated -->
            <BurstTime>1</BurstTime> <!-- Time the beam will fire -->
            <InitialTicks>0</InitialTicks> <!-- Ticks to do at the initial contact of beam with a block -->
            <AdditionalPowerConsumptionPerUnitMult>0.1</AdditionalPowerConsumptionPerUnitMult> <!-- Nerf based on amount of groups connected to the same controller: powerConsumption * (1+countGroups*thisValue); 0 to turn off nerf-->
            <DefaultWeaponPenetrationWeight>1.5</DefaultWeaponPenetrationWeight> <!-- This weight will distribute the damage on the penetration depth path. The higher, the more damage is used in the first blocks -->

<DefaultWeaponPenetrationDeepnessMultiplier>1</DefaultWeaponPenetrationDeepnessMultiplier> <!-- how deep is the penetration potentially (depending on log10 of damage). A value of 0 defaults to the old system (full damage hits one block) -->
            <RailHitMultiplierParent>0</RailHitMultiplierParent> <!-- Damage percent done to target in own Rail Chain when target is equal or closer to root than this entitiy-->
            <RailHitMultiplierChild>0</RailHitMultiplierChild> <!-- Damage percent done to target in own Rail Chain when target is further from root than this entity -->
        </BasicValues>
        <Combination>
            <Cannon>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
                <Distance style="skip" />
                <PowerConsumption style="buff" inverse="true" value="5" linear="true" />
                <PowerPerHit style="nerf" value="5" linear="true" /> <!-- This is the damage per hit -->
                <Split style="skip" />
                <CoolDown style="buff" inverse="true" value="5"/> <!-- Time it takes to fire beam again from the start-time it first activated -->
                <BurstTime style="skip"/> <!-- Time the beam will fire -->
                <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
            </Cannon>
            <Missile>
                <HitSpeed style="skip" /> <!-- Affects TickRate -->
```

```
                    <Distance style="skip" />
                    <PowerConsumption style="skip" linear="true" />
                    <PowerPerHit style="skip" linear="true"/> <!-- linear is needed here -->
                    <Split style="buff" value="9" />
                    <CoolDown style="skip"/> <!-- Time it takes to fire beam again from the start-time it first activated
-->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Missile>
                <Beam>
                    <HitSpeed style="skip" /> <!-- Affects TickRate -->
                    <Distance style="buff" value="2" />
                    <PowerConsumption style="nerf" inverse="true" linear="true" value="2" />
                    <PowerPerHit style="buff" linear="true" value="2" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="2"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Beam>
                <Pulse>
                    <HitSpeed style="skip"/> <!-- Affects TickRate -->
                    <Distance style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="10" linear="true" />
                    <PowerPerHit style="buff" value="10" linear="true" />
                    <Split style="skip" />
                    <CoolDown style="nerf" inverse="true" value="10.0"/> <!-- Time it takes to fire beam again from the
start-time it first activated -->
                    <BurstTime style="skip"/> <!-- Time the beam will fire -->
                    <InitialTicks style="skip"/> <!-- Ticks to do at the initial contact of beam with a block -->
                </Pulse>
            </Combination>
        </DamageBeam>
        <DamagePulse>
            <BasicValues>
                <Damage>100</Damage>  <!-- Uses the missile damage system, detonation point is <Radius> blocks
away from output block -->
                <ReloadMs>10000</ReloadMs>
                <Radius>10</Radius>
                <PowerConsumption>500</PowerConsumption> <!-- deviates from 10 power/1damage formula for now,
it's hard to use this weapon -->
                <AdditionalPowerConsumptionPerUnitMult>0.0</AdditionalPowerConsumptionPerUnitMult> <!-- Nerf
based on amount of groups connected to the same controller: powerConsumption * (1+countGroups*thisValue); 0 to
turn off nerf-->
            </BasicValues>
            <Combination>
```

```
                <Cannon>
                    <PulsePower style="nerf" value="5" linear="true" />             <!-- this is the same as PowerPerHit
-->
                    <Reload style="buff" inverse="true" value="5" />
                    <Radius style="skip" />
                    <PowerConsumption style="nerf" value="5" linear="true" />
                </Cannon>
                <Missile>
                    <PulsePower style="nerf" value="5" linear="true" />           <!-- since the pulse is a localized burst,
creating multiples overlapping is pointless, so we simply mimic the cannon slaved effect and increase rate of fire -->
                    <Reload style="buff" inverse="true" value="5" />  <!-- Ideally this variation should drop a single
munition that pulses 5 times where it is dropped -->
                    <Radius style="skip" />
                    <PowerConsumption style="nerf" value="5" linear="true" />
                </Missile>
                <Beam>
                    <PulsePower style="buff" value="2" linear="true" />
                    <Reload style="nerf" inverse="true" value="2" />
                    <Radius style="buff" value="1.5" />
                    <PowerConsumption style="nerf" inverse="true" value="2" linear="true" />
                </Beam>
                <Pulse>
                    <PulsePower style="buff" value="5" linear="true" />
                    <Reload style="nerf" inverse="true" value="5" />
                    <Radius style="buff" value="3" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                </Pulse>
            </Combination>
        </DamagePulse>
        <PushPulse>
            <BasicValues>
                <Force>25</Force>
                <ReloadMs>5000</ReloadMs>
                <Radius>100</Radius>
                <PowerConsumption>250</PowerConsumption>
            </BasicValues>
            <Combination>
                <Cannon>
                    <PulsePower style="nerf" value="5" linear="true" />
                    <Reload style="buff" inverse="true" value="5" />
                    <Radius style="skip" />
                    <PowerConsumption style="nerf" value="5" linear="true" />
                </Cannon>
                <Missile>
                    <PulsePower style="nerf" value="5" linear="true" />           <!-- since the pulse is a localized burst,
creating multiples overlapping is pointless, so we simply mimic the cannon slaved effect and increase rate of fire -->
```

```
                    <Reload style="buff"  inverse="true" value="5"/> <!-- Ideally this variation should drop a single
munition that pulses 5 times where it is dropped -->
                    <Radius style="skip" />
                    <PowerConsumption style="nerf" value="5" linear="true" />
                </Missile>
                <Beam>
                    <PulsePower style="skip" linear="true" />
                    <Reload style="skip"/>
                    <Radius style="buff" value="1.5" />
                    <PowerConsumption style="nerf" inverse="true" value="2" linear="true" />
                </Beam>
                <Pulse>
                    <PulsePower style="buff" value="5" linear="true" />
                    <Reload style="nerf" inverse="true" value="5" />
                    <Radius style="skip" />
                    <PowerConsumption style="nerf" inverse="true" value="5" linear="true" />
                </Pulse>
            </Combination>
        </PushPulse>

        <!-- defensive status effect:

STATUS_ARMOR_HARDEN,STATUS_PIERCING_PROTECTION,STATUS_POWER_SHIELD,

STATUS_SHIELD_HARDEN,STATUS_TOP_SPEED,STATUS_ANTI_GRAVITY,STATUS_GRAVITY_EFFECT_IGNOR
ANCE,TAKE_OFF,EVADE ,
                                                    STATUS_ARMOR_HP_ABSORBTION_BONUS,
STATUS_ARMOR_HP_DEDUCTION_BONUS
                        all defensive status effects (STATUS_...) are calculated with
                            effectEfficiency = (effectBlockCount*multiplier) / totalBlockCount
                    -->
        <EMP>
            <BasicValues>
                <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <CannonBlockDamageBonus>-1</CannonBlockDamageBonus>
                <CannonShieldDamageBonus>-1</CannonShieldDamageBonus>
                <CannonPowerDamageBonus>20</CannonPowerDamageBonus>
                <CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
                <CannonArmorEfficiency>0</CannonArmorEfficiency>
                <CannonPullEffect>0</CannonPullEffect>
                <CannonPushEffect>0</CannonPushEffect>
                <CannonGrabEffect>0</CannonGrabEffect>
                <CannonSystemHPBonus>0</CannonSystemHPBonus>
                <CannonArmorHPBonus>0</CannonArmorHPBonus>
```

<MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<MissileBlockDamageBonus>-1</MissileBlockDamageBonus>
<MissileShieldDamageBonus>-1</MissileShieldDamageBonus>
<MissilePowerDamageBonus>20</MissilePowerDamageBonus>
<MissileExplosiveRadius>0</MissileExplosiveRadius>
<MissileArmorEfficiency>0</MissileArmorEfficiency>
<MissilePullEffect>0</MissilePullEffect>
<MissilePushEffect>0</MissilePushEffect>
<MissileGrabEffect>0</MissileGrabEffect>
<MissileSystemHPBonus>0</MissileSystemHPBonus>
<MissileArmorHPBonus>0</MissileArmorHPBonus>
<MissileTotalDamageBonus>0</MissileTotalDamageBonus>

<PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<PulseBlockDamageBonus>-1</PulseBlockDamageBonus>
<PulseShieldDamageBonus>-1</PulseShieldDamageBonus>
<PulsePowerDamageBonus>20</PulsePowerDamageBonus>
<PulseExplosiveRadius>0</PulseExplosiveRadius>
<PulseArmorEfficiency>0</PulseArmorEfficiency>
<PulsePullEffect>0</PulsePullEffect>
<PulsePushEffect>0</PulsePushEffect>
<PulseGrabEffect>0</PulseGrabEffect>
<PulseSystemHPBonus>0</PulseSystemHPBonus>
<PulseArmorHPBonus>0</PulseArmorHPBonus>
<PulseTotalDamageBonus>0</PulseTotalDamageBonus>

<BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<BeamBlockDamageBonus>-1</BeamBlockDamageBonus>
<BeamShieldDamageBonus>-1</BeamShieldDamageBonus>
<BeamPowerDamageBonus>20</BeamPowerDamageBonus>
<BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented later-->
<BeamArmorEfficiency>0</BeamArmorEfficiency>
<BeamPullEffect>0</BeamPullEffect>
<BeamPushEffect>0</BeamPushEffect>
<BeamGrabEffect>0</BeamGrabEffect>
<BeamSystemHPBonus>0</BeamSystemHPBonus>
<BeamArmorHPBonus>0</BeamArmorHPBonus>

<Piercing>false</Piercing>
<PunchThrough>false</PunchThrough>

```
                <Explosive>false</Explosive>
                <PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
                <PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
                <DefensiveBasePowerConsumption>10</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
                <DefensiveBaseMultiplier>10</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->

<DefensiveEffectType>STATUS_POWER_SHIELD</DefensiveEffectType>
                                        <EffectConnectDescription>The weapon's damage is forwarded to
the\npower system of the target.</EffectConnectDescription>
                                        <EffectIgnoresShields>true</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>0.5</DefensiveEffectCapPercent>



            </BasicValues>
        </EMP>
        <PunchThrough>
            <BasicValues>
                <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <CannonBlockDamageBonus>0.5</CannonBlockDamageBonus>
                <CannonShieldDamageBonus>0</CannonShieldDamageBonus>
                <CannonPowerDamageBonus>0</CannonPowerDamageBonus>
                <CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
                <CannonArmorEfficiency>0</CannonArmorEfficiency>
                <CannonPullEffect>0</CannonPullEffect>
                <CannonPushEffect>0</CannonPushEffect>
                <CannonGrabEffect>0</CannonGrabEffect>
                <CannonSystemHPBonus>0</CannonSystemHPBonus>
                <CannonArmorHPBonus>0.25</CannonArmorHPBonus>


                <MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <MissileBlockDamageBonus>0.0</MissileBlockDamageBonus>
                <MissileShieldDamageBonus>0</MissileShieldDamageBonus>
                <MissilePowerDamageBonus>0</MissilePowerDamageBonus>
                <MissileExplosiveRadius>5</MissileExplosiveRadius>
                <MissileArmorEfficiency>0</MissileArmorEfficiency>
                <MissilePullEffect>0</MissilePullEffect>
                <MissilePushEffect>0</MissilePushEffect>
                <MissileGrabEffect>0</MissileGrabEffect>
                <MissileSystemHPBonus>0</MissileSystemHPBonus>
                <MissileArmorHPBonus>0.5</MissileArmorHPBonus>
```

```
		<MissileTotalDamageBonus>0</MissileTotalDamageBonus>

		<PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<PulseBlockDamageBonus>0.0</PulseBlockDamageBonus>
		<PulseShieldDamageBonus>0</PulseShieldDamageBonus>
		<PulsePowerDamageBonus>0</PulsePowerDamageBonus>
		<PulseExplosiveRadius>5</PulseExplosiveRadius>
		<PulseArmorEfficiency>0</PulseArmorEfficiency>
		<PulsePullEffect>0</PulsePullEffect>
		<PulsePushEffect>0</PulsePushEffect>
		<PulseGrabEffect>0</PulseGrabEffect>
		<PulseSystemHPBonus>0</PulseSystemHPBonus>
		<PulseArmorHPBonus>0.5</PulseArmorHPBonus>
		<PulseTotalDamageBonus>0</PulseTotalDamageBonus>

		<BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<BeamBlockDamageBonus>0.5</BeamBlockDamageBonus>
		<BeamShieldDamageBonus>0</BeamShieldDamageBonus>
		<BeamPowerDamageBonus>0</BeamPowerDamageBonus>
		<BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
		<BeamArmorEfficiency>0</BeamArmorEfficiency>
		<BeamPullEffect>0</BeamPullEffect>
		<BeamPushEffect>0</BeamPushEffect>
		<BeamGrabEffect>0</BeamGrabEffect>
		<BeamSystemHPBonus>0</BeamSystemHPBonus>
		<BeamArmorHPBonus>0.25</BeamArmorHPBonus>


		<Piercing>false</Piercing>
		<PunchThrough>false</PunchThrough>
		<Explosive>false</Explosive>
		<PiercingDamagePreservedOnImpact>0.0</PiercingDamagePreservedOnImpact>
		<PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
		<DefensiveBasePowerConsumption>10</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
		<DefensiveBaseMultiplier>45</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->

<DefensiveEffectType>STATUS_ARMOR_HP_DEDUCTION_BONUS</DefensiveEffectType>
			<EffectConnectDescription></EffectConnectDescription>
			<EffectIgnoresShields>false</EffectIgnoresShields>
			<DefensiveEffectCapPercent>0.25</DefensiveEffectCapPercent>
	</BasicValues>
</PunchThrough>
```

```
    <Piercing>
        <BasicValues>
            <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <CannonBlockDamageBonus>0.25</CannonBlockDamageBonus>
            <CannonShieldDamageBonus>-1</CannonShieldDamageBonus>
            <CannonPowerDamageBonus>0</CannonPowerDamageBonus>
            <CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
            <CannonArmorEfficiency>-0.5</CannonArmorEfficiency>
            <CannonPullEffect>0</CannonPullEffect>
            <CannonPushEffect>0</CannonPushEffect>
            <CannonGrabEffect>0</CannonGrabEffect>
            <CannonSystemHPBonus>0</CannonSystemHPBonus>
            <CannonArmorHPBonus>0.5</CannonArmorHPBonus>


            <MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <MissileBlockDamageBonus>0</MissileBlockDamageBonus>
            <MissileShieldDamageBonus>0</MissileShieldDamageBonus>
            <MissilePowerDamageBonus>0</MissilePowerDamageBonus>
            <MissileExplosiveRadius>5</MissileExplosiveRadius>
            <MissileArmorEfficiency>-0.5</MissileArmorEfficiency>
            <MissilePullEffect>0</MissilePullEffect>
            <MissilePushEffect>0</MissilePushEffect>
            <MissileGrabEffect>0</MissileGrabEffect>
            <MissileSystemHPBonus>0</MissileSystemHPBonus>
            <MissileArmorHPBonus>0</MissileArmorHPBonus>
            <MissileTotalDamageBonus>0</MissileTotalDamageBonus>

            <PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <PulseBlockDamageBonus>0</PulseBlockDamageBonus>
            <PulseShieldDamageBonus>0</PulseShieldDamageBonus>
            <PulsePowerDamageBonus>0</PulsePowerDamageBonus>
            <PulseExplosiveRadius>5</PulseExplosiveRadius>
            <PulseArmorEfficiency>-0.5</PulseArmorEfficiency>
            <PulsePullEffect>0</PulsePullEffect>
            <PulsePushEffect>0</PulsePushEffect>
            <PulseGrabEffect>0</PulseGrabEffect>
            <PulseSystemHPBonus>0</PulseSystemHPBonus>
            <PulseArmorHPBonus>0</PulseArmorHPBonus>
            <PulseTotalDamageBonus>0</PulseTotalDamageBonus>
```

```
                <BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <BeamBlockDamageBonus>0.25</BeamBlockDamageBonus>
                <BeamShieldDamageBonus>-1</BeamShieldDamageBonus>
                <BeamPowerDamageBonus>0</BeamPowerDamageBonus>
                <BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
                <BeamArmorEfficiency>-0.5</BeamArmorEfficiency>
                <BeamPullEffect>0</BeamPullEffect>
                <BeamPushEffect>0</BeamPushEffect>
                <BeamGrabEffect>0</BeamGrabEffect>
                <BeamSystemHPBonus>0</BeamSystemHPBonus>
                <BeamArmorHPBonus>0.5</BeamArmorHPBonus>



                <Piercing>false</Piercing>
                <PunchThrough>false</PunchThrough>
                <Explosive>false</Explosive>
                <PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
                <PunchThroughDamagePreserved>0.0</PunchThroughDamagePreserved>
                <DefensiveBasePowerConsumption>10</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
                <DefensiveBaseMultiplier>45</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->

<DefensiveEffectType>STATUS_ARMOR_HP_ABSORPTION_BONUS</DefensiveEffectType>
                                        <EffectConnectDescription></EffectConnectDescription>
                                        <EffectIgnoresShields>false</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>0.25</DefensiveEffectCapPercent>
        </BasicValues>
    </Piercing>
    <ExplosionEffect>
        <BasicValues>
                <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <CannonBlockDamageBonus>-0.5</CannonBlockDamageBonus>
                <CannonShieldDamageBonus>0</CannonShieldDamageBonus>
                <CannonPowerDamageBonus>0</CannonPowerDamageBonus>
                <CannonExplosiveRadius>10</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
                <CannonArmorEfficiency>0</CannonArmorEfficiency>
                <CannonPullEffect>0</CannonPullEffect>
                <CannonPushEffect>0</CannonPushEffect>
                <CannonGrabEffect>0</CannonGrabEffect>
                <CannonSystemHPBonus>0</CannonSystemHPBonus>
                <CannonArmorHPBonus>0</CannonArmorHPBonus>
```

```
		<MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<MissileBlockDamageBonus>0</MissileBlockDamageBonus>
		<MissileShieldDamageBonus>0</MissileShieldDamageBonus>
		<MissilePowerDamageBonus>0</MissilePowerDamageBonus>
		<MissileExplosiveRadius>10</MissileExplosiveRadius> <!-- adds radius to the weapon -->
		<MissileArmorEfficiency>0</MissileArmorEfficiency>
		<MissilePullEffect>0</MissilePullEffect>
		<MissilePushEffect>0</MissilePushEffect>
		<MissileGrabEffect>0</MissileGrabEffect>
		<MissileSystemHPBonus>0</MissileSystemHPBonus>
		<MissileArmorHPBonus>0</MissileArmorHPBonus>
		<MissileTotalDamageBonus>0</MissileTotalDamageBonus>


		<PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<PulseBlockDamageBonus>0</PulseBlockDamageBonus>
		<PulseShieldDamageBonus>0</PulseShieldDamageBonus>
		<PulsePowerDamageBonus>0</PulsePowerDamageBonus>
		<PulseExplosiveRadius>10</PulseExplosiveRadius> <!-- adds radius to the weapon -->
		<PulseArmorEfficiency>0</PulseArmorEfficiency>
		<PulsePullEffect>0</PulsePullEffect>
		<PulsePushEffect>0</PulsePushEffect>
		<PulseGrabEffect>0</PulseGrabEffect>
		<PulseSystemHPBonus>0</PulseSystemHPBonus>
		<PulseArmorHPBonus>0</PulseArmorHPBonus>
		<PulseTotalDamageBonus>0</PulseTotalDamageBonus>

		<BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<BeamBlockDamageBonus>-0.5</BeamBlockDamageBonus>
		<BeamShieldDamageBonus>0</BeamShieldDamageBonus>
		<BeamPowerDamageBonus>0</BeamPowerDamageBonus>
		<BeamExplosiveRadius>10</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
		<BeamArmorEfficiency>0</BeamArmorEfficiency>
		<BeamPullEffect>0</BeamPullEffect>
		<BeamPushEffect>0</BeamPushEffect>
		<BeamGrabEffect>0</BeamGrabEffect>
		<BeamSystemHPBonus>0</BeamSystemHPBonus>
		<BeamArmorHPBonus>0</BeamArmorHPBonus>


		<Piercing>false</Piercing>
```

<PunchThrough>false</PunchThrough>
<Explosive>true</Explosive>
<PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
<PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
<DefensiveBasePowerConsumption>15</DefensiveBasePowerConsumption> <!-- Power cost per block per second, % bonus is based on the system/ship mass size ratio -->
<DefensiveBaseMultiplier>40</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->

<DefensiveEffectType>STATUS_GRAVITY_EFFECT_IGNORANCE</DefensiveEffectType>
<EffectConnectDescription>Adds explosion effect to the weapon\nhitting up to 6 blocks at once.\nAdds extra radius to missiles and pulse.</EffectConnectDescription>
<EffectIgnoresShields>false</EffectIgnoresShields>
<DefensiveEffectCapPercent>1.0</DefensiveEffectCapPercent>
</BasicValues>
</ExplosionEffect>
<StopEffect>
<BasicValues>
<CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<CannonBlockDamageBonus>-1</CannonBlockDamageBonus>
<CannonShieldDamageBonus>-1</CannonShieldDamageBonus>
<CannonPowerDamageBonus>0</CannonPowerDamageBonus>
<CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be implemented later-->
<CannonArmorEfficiency>0</CannonArmorEfficiency>
<CannonPullEffect>0</CannonPullEffect>
<CannonPushEffect>0</CannonPushEffect>
<CannonGrabEffect>2</CannonGrabEffect>
<CannonSystemHPBonus>0</CannonSystemHPBonus>
<CannonArmorHPBonus>0</CannonArmorHPBonus>

<MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<MissileBlockDamageBonus>-1</MissileBlockDamageBonus>
<MissileShieldDamageBonus>-1</MissileShieldDamageBonus>
<MissilePowerDamageBonus>0</MissilePowerDamageBonus>
<MissileExplosiveRadius>0</MissileExplosiveRadius>
<MissileArmorEfficiency>0</MissileArmorEfficiency>
<MissilePullEffect>0</MissilePullEffect>
<MissilePushEffect>0</MissilePushEffect>
<MissileGrabEffect>2</MissileGrabEffect>
<MissileSystemHPBonus>0</MissileSystemHPBonus>
<MissileArmorHPBonus>0</MissileArmorHPBonus>
<MissileTotalDamageBonus>0</MissileTotalDamageBonus>

```
                <PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <PulseBlockDamageBonus>-1</PulseBlockDamageBonus>
                <PulseShieldDamageBonus>-1</PulseShieldDamageBonus>
                <PulsePowerDamageBonus>0</PulsePowerDamageBonus>
                <PulseExplosiveRadius>0</PulseExplosiveRadius>
                <PulseArmorEfficiency>0</PulseArmorEfficiency>
                <PulsePullEffect>0</PulsePullEffect>
                <PulsePushEffect>0</PulsePushEffect>
                <PulseGrabEffect>2</PulseGrabEffect>
                <PulseSystemHPBonus>0</PulseSystemHPBonus>
                <PulseArmorHPBonus>0</PulseArmorHPBonus>
                <PulseTotalDamageBonus>0</PulseTotalDamageBonus>

                <BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <BeamBlockDamageBonus>-1</BeamBlockDamageBonus>
                <BeamShieldDamageBonus>-1</BeamShieldDamageBonus>
                <BeamPowerDamageBonus>0</BeamPowerDamageBonus>
                <BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
                <BeamArmorEfficiency>0</BeamArmorEfficiency>
                <BeamPullEffect>0</BeamPullEffect>
                <BeamPushEffect>0</BeamPushEffect>
                <BeamGrabEffect>2</BeamGrabEffect>
                <BeamSystemHPBonus>0</BeamSystemHPBonus>
                <BeamArmorHPBonus>0</BeamArmorHPBonus>



                <Piercing>false</Piercing>
                <PunchThrough>false</PunchThrough>
                <Explosive>false</Explosive>
                <PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
                <PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
                <DefensiveBasePowerConsumption>10</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
                <DefensiveBaseMultiplier>40</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->
                                        <DefensiveEffectType>STATUS_ANTI_GRAVITY</DefensiveEffectType>
                                        <EffectConnectDescription>Adds breaking effect to the
weapon</EffectConnectDescription>
                                        <EffectIgnoresShields>true</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>1</DefensiveEffectCapPercent>
        </BasicValues>
    </StopEffect>
    <PushEffect>
        <BasicValues>
```

<CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<CannonBlockDamageBonus>-1</CannonBlockDamageBonus>
<CannonShieldDamageBonus>-1</CannonShieldDamageBonus>
<CannonPowerDamageBonus>0</CannonPowerDamageBonus>
<CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be implemented later-->
<CannonArmorEfficiency>0</CannonArmorEfficiency>
<CannonPullEffect>0</CannonPullEffect>
<CannonPushEffect>2</CannonPushEffect>
<CannonGrabEffect>0</CannonGrabEffect>
<CannonSystemHPBonus>0</CannonSystemHPBonus>
<CannonArmorHPBonus>0</CannonArmorHPBonus>

<MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<MissileBlockDamageBonus>-1</MissileBlockDamageBonus>
<MissileShieldDamageBonus>-1</MissileShieldDamageBonus>
<MissilePowerDamageBonus>0</MissilePowerDamageBonus>
<MissileExplosiveRadius>0</MissileExplosiveRadius>
<MissileArmorEfficiency>0</MissileArmorEfficiency>
<MissilePullEffect>0</MissilePullEffect>
<MissilePushEffect>2</MissilePushEffect>
<MissileGrabEffect>0</MissileGrabEffect>
<MissileSystemHPBonus>0</MissileSystemHPBonus>
<MissileArmorHPBonus>0</MissileArmorHPBonus>
<MissileTotalDamageBonus>0</MissileTotalDamageBonus>

<PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<PulseBlockDamageBonus>-1</PulseBlockDamageBonus>
<PulseShieldDamageBonus>-1</PulseShieldDamageBonus>
<PulsePowerDamageBonus>0</PulsePowerDamageBonus>
<PulseExplosiveRadius>0</PulseExplosiveRadius>
<PulseArmorEfficiency>0</PulseArmorEfficiency>
<PulsePullEffect>0</PulsePullEffect>
<PulsePushEffect>2</PulsePushEffect>
<PulseGrabEffect>0</PulseGrabEffect>
<PulseSystemHPBonus>0</PulseSystemHPBonus>
<PulseArmorHPBonus>0</PulseArmorHPBonus>
<PulseTotalDamageBonus>0</PulseTotalDamageBonus>

<BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<BeamBlockDamageBonus>-1</BeamBlockDamageBonus>

```
                <BeamShieldDamageBonus>-1</BeamShieldDamageBonus>
                <BeamPowerDamageBonus>0</BeamPowerDamageBonus>
                <BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
                <BeamArmorEfficiency>0</BeamArmorEfficiency>
                <BeamPullEffect>0</BeamPullEffect>
                <BeamPushEffect>2</BeamPushEffect>
                <BeamGrabEffect>0</BeamGrabEffect>
                <BeamSystemHPBonus>0</BeamSystemHPBonus>
                <BeamArmorHPBonus>0</BeamArmorHPBonus>


                <Piercing>false</Piercing>
                <PunchThrough>false</PunchThrough>
                <Explosive>false</Explosive>
                <PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
                <PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
                <DefensiveBasePowerConsumption>15</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
                <DefensiveBaseMultiplier>20</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->
                                        <DefensiveEffectType>TAKE_OFF</DefensiveEffectType>
                                        <EffectConnectDescription>Adds push effect to the
weapon</EffectConnectDescription>
                                        <EffectIgnoresShields>true</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>1.0</DefensiveEffectCapPercent>
        </BasicValues>
    </PushEffect>
    <PullEffect>
        <BasicValues>
            <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <CannonBlockDamageBonus>-1</CannonBlockDamageBonus>
                <CannonShieldDamageBonus>-1</CannonShieldDamageBonus>
                <CannonPowerDamageBonus>0</CannonPowerDamageBonus>
                <CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
                <CannonArmorEfficiency>0</CannonArmorEfficiency>
                <CannonPullEffect>2</CannonPullEffect>
                <CannonPushEffect>0</CannonPushEffect>
                <CannonGrabEffect>0</CannonGrabEffect>
                <CannonSystemHPBonus>0</CannonSystemHPBonus>
                <CannonArmorHPBonus>0</CannonArmorHPBonus>


                <MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
```

```
		<MissileBlockDamageBonus>-1</MissileBlockDamageBonus>
		<MissileShieldDamageBonus>-1</MissileShieldDamageBonus>
		<MissilePowerDamageBonus>0</MissilePowerDamageBonus>
		<MissileExplosiveRadius>0</MissileExplosiveRadius>
		<MissileArmorEfficiency>0</MissileArmorEfficiency>
		<MissilePullEffect>2</MissilePullEffect>
		<MissilePushEffect>0</MissilePushEffect>
		<MissileGrabEffect>0</MissileGrabEffect>
		<MissileSystemHPBonus>0</MissileSystemHPBonus>
		<MissileArmorHPBonus>0</MissileArmorHPBonus>
		<MissileTotalDamageBonus>0</MissileTotalDamageBonus>

		<PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<PulseBlockDamageBonus>-1</PulseBlockDamageBonus>
		<PulseShieldDamageBonus>-1</PulseShieldDamageBonus>
		<PulsePowerDamageBonus>0</PulsePowerDamageBonus>
		<PulseExplosiveRadius>0</PulseExplosiveRadius>
		<PulseArmorEfficiency>0</PulseArmorEfficiency>
		<PulsePullEffect>2</PulsePullEffect>
		<PulsePushEffect>0</PulsePushEffect>
		<PulseGrabEffect>0</PulseGrabEffect>
		<PulseSystemHPBonus>0</PulseSystemHPBonus>
		<PulseArmorHPBonus>0</PulseArmorHPBonus>
		<PulseTotalDamageBonus>0</PulseTotalDamageBonus>

		<BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
		<BeamBlockDamageBonus>-1</BeamBlockDamageBonus>
		<BeamShieldDamageBonus>-1</BeamShieldDamageBonus>
		<BeamPowerDamageBonus>0</BeamPowerDamageBonus>
		<BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
		<BeamArmorEfficiency>0</BeamArmorEfficiency>
		<BeamPullEffect>2</BeamPullEffect>
		<BeamPushEffect>0</BeamPushEffect>
		<BeamGrabEffect>0</BeamGrabEffect>
		<BeamSystemHPBonus>0</BeamSystemHPBonus>
		<BeamArmorHPBonus>0</BeamArmorHPBonus>


		<Piercing>false</Piercing>
		<PunchThrough>false</PunchThrough>
		<Explosive>false</Explosive>
		<PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
		<PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
```

```
            <DefensiveBasePowerConsumption>15</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
            <DefensiveBaseMultiplier>20</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->
                                        <DefensiveEffectType>EVADE</DefensiveEffectType>
                                        <EffectConnectDescription>Adds pull effect to the
weapon</EffectConnectDescription>
                                        <EffectIgnoresShields>true</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>1.0</DefensiveEffectCapPercent>
        </BasicValues>
    </PullEffect>
    <IonEffect>
        <BasicValues>
            <CannonBasePowerConsumption>0</CannonBasePowerConsumption> <!-- power/shot cost
additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <CannonBlockDamageBonus>-1</CannonBlockDamageBonus>
            <CannonShieldDamageBonus>1</CannonShieldDamageBonus>
            <CannonPowerDamageBonus>0</CannonPowerDamageBonus>
            <CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be
implemented later-->
            <CannonArmorEfficiency>0</CannonArmorEfficiency>
            <CannonPullEffect>0</CannonPullEffect>
            <CannonPushEffect>0</CannonPushEffect>
            <CannonGrabEffect>0</CannonGrabEffect>
            <CannonSystemHPBonus>0</CannonSystemHPBonus>
            <CannonArmorHPBonus>0</CannonArmorHPBonus>


            <MissileBasePowerConsumption>0</MissileBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <MissileBlockDamageBonus>-1</MissileBlockDamageBonus>
            <MissileShieldDamageBonus>0.5</MissileShieldDamageBonus> <!-- shield damage is doubled, the
explosion system scales differently-->
            <MissilePowerDamageBonus>0</MissilePowerDamageBonus>
            <MissileExplosiveRadius>0</MissileExplosiveRadius>
            <MissileArmorEfficiency>0</MissileArmorEfficiency>
            <MissilePullEffect>0</MissilePullEffect>
            <MissilePushEffect>0</MissilePushEffect>
            <MissileGrabEffect>0</MissileGrabEffect>
            <MissileSystemHPBonus>0</MissileSystemHPBonus>
            <MissileArmorHPBonus>0</MissileArmorHPBonus>
            <MissileTotalDamageBonus>0</MissileTotalDamageBonus>

            <PulseBasePowerConsumption>0</PulseBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
            <PulseBlockDamageBonus>-1</PulseBlockDamageBonus>
```

```
                <PulseShieldDamageBonus>0.5</PulseShieldDamageBonus> <!-- shield damage is doubled, the
explosion system scales differently-->
                <PulsePowerDamageBonus>0</PulsePowerDamageBonus>
                <PulseExplosiveRadius>0</PulseExplosiveRadius>
                <PulseArmorEfficiency>0</PulseArmorEfficiency>
                <PulsePullEffect>0</PulsePullEffect>
                <PulsePushEffect>0</PulsePushEffect>
                <PulseGrabEffect>0</PulseGrabEffect>
                <PulseSystemHPBonus>0</PulseSystemHPBonus>
                <PulseArmorHPBonus>0</PulseArmorHPBonus>
                <PulseTotalDamageBonus>0</PulseTotalDamageBonus>

                <BeamBasePowerConsumption>0</BeamBasePowerConsumption> <!-- power/shot cost additionally
added (normalCost+(normalCost*thisValue*effectRatio)) -->
                <BeamBlockDamageBonus>-1</BeamBlockDamageBonus>
                <BeamShieldDamageBonus>1</BeamShieldDamageBonus>
                <BeamPowerDamageBonus>0</BeamPowerDamageBonus>
                <BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented
later-->
                <BeamArmorEfficiency>0</BeamArmorEfficiency>
                <BeamPullEffect>0</BeamPullEffect>
                <BeamPushEffect>0</BeamPushEffect>
                <BeamGrabEffect>0</BeamGrabEffect>
                <BeamSystemHPBonus>0</BeamSystemHPBonus>
                <BeamArmorHPBonus>0</BeamArmorHPBonus>



                <Piercing>false</Piercing>
                <PunchThrough>false</PunchThrough>
                <Explosive>false</Explosive>
                <PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
                <PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
                <DefensiveBasePowerConsumption>20</DefensiveBasePowerConsumption>  <!-- Power cost per
block per second, % bonus is based on the system/ship mass size ratio -->
                 <DefensiveBaseMultiplier>12</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->

<DefensiveEffectType>STATUS_SHIELD_HARDEN</DefensiveEffectType>
                                        <EffectConnectDescription>Does extra damage against shields,\nbut less
against blocks</EffectConnectDescription>
                                        <EffectIgnoresShields>false</EffectIgnoresShields>
                                        <DefensiveEffectCapPercent>0.6</DefensiveEffectCapPercent>
        </BasicValues>
    </IonEffect>
    <OverdriveEffect>
        <BasicValues>
```

<CannonBasePowerConsumption>5</CannonBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<CannonBlockDamageBonus>2</CannonBlockDamageBonus>
<CannonShieldDamageBonus>2</CannonShieldDamageBonus>
<CannonPowerDamageBonus>0</CannonPowerDamageBonus>
<CannonExplosiveRadius>0</CannonExplosiveRadius> <!-- currently capped at 1. may be implemented later-->
<CannonArmorEfficiency>0</CannonArmorEfficiency>
<CannonPullEffect>0</CannonPullEffect>
<CannonPushEffect>0</CannonPushEffect>
<CannonGrabEffect>0</CannonGrabEffect>
<CannonSystemHPBonus>0</CannonSystemHPBonus>
<CannonArmorHPBonus>0</CannonArmorHPBonus>

<MissileBasePowerConsumption>5</MissileBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<MissileBlockDamageBonus>0</MissileBlockDamageBonus>
<MissileShieldDamageBonus>0</MissileShieldDamageBonus>
<MissilePowerDamageBonus>0</MissilePowerDamageBonus>
<MissileExplosiveRadius>10</MissileExplosiveRadius>
<MissileArmorEfficiency>0</MissileArmorEfficiency>
<MissilePullEffect>0</MissilePullEffect>
<MissilePushEffect>0</MissilePushEffect>
<MissileGrabEffect>0</MissileGrabEffect>
<MissileSystemHPBonus>0</MissileSystemHPBonus>
<MissileArmorHPBonus>0</MissileArmorHPBonus>
<MissileTotalDamageBonus>2</MissileTotalDamageBonus>

<PulseBasePowerConsumption>5</PulseBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<PulseBlockDamageBonus>0</PulseBlockDamageBonus>
<PulseShieldDamageBonus>0</PulseShieldDamageBonus>
<PulsePowerDamageBonus>0</PulsePowerDamageBonus>
<PulseExplosiveRadius>10</PulseExplosiveRadius>
<PulseArmorEfficiency>0</PulseArmorEfficiency>
<PulsePullEffect>0</PulsePullEffect>
<PulsePushEffect>0</PulsePushEffect>
<PulseGrabEffect>0</PulseGrabEffect>
<PulseSystemHPBonus>0</PulseSystemHPBonus>
<PulseArmorHPBonus>0</PulseArmorHPBonus>
<PulseTotalDamageBonus>2</PulseTotalDamageBonus>

<BeamBasePowerConsumption>5</BeamBasePowerConsumption> <!-- power/shot cost additionally added (normalCost+(normalCost*thisValue*effectRatio)) -->
<BeamBlockDamageBonus>2</BeamBlockDamageBonus>

<BeamShieldDamageBonus>2</BeamShieldDamageBonus>
<BeamPowerDamageBonus>0</BeamPowerDamageBonus>
<BeamExplosiveRadius>0</BeamExplosiveRadius> <!-- currently capped at 1. may be implemented later-->
<BeamArmorEfficiency>0</BeamArmorEfficiency>
<BeamPullEffect>0</BeamPullEffect>
<BeamPushEffect>0</BeamPushEffect>
<BeamGrabEffect>0</BeamGrabEffect>
<BeamSystemHPBonus>0</BeamSystemHPBonus>
<BeamArmorHPBonus>0</BeamArmorHPBonus>

<Piercing>false</Piercing>
<PunchThrough>false</PunchThrough>
<Explosive>false</Explosive>
<PiercingDamagePreservedOnImpact>0</PiercingDamagePreservedOnImpact>
<PunchThroughDamagePreserved>0</PunchThroughDamagePreserved>
<DefensiveBasePowerConsumption>10</DefensiveBasePowerConsumption> <!-- Power cost per block per second, % bonus is based on the system/ship mass size ratio -->
<DefensiveBaseMultiplier>20</DefensiveBaseMultiplier> <!-- multiplied with effect block count -->
<DefensiveEffectType>STATUS_TOP_SPEED</DefensiveEffectType>
<EffectConnectDescription>More damage for power usage</EffectConnectDescription>
<EffectIgnoresShields>false</EffectIgnoresShields>
<DefensiveEffectCapPercent>0.5</DefensiveEffectCapPercent>
</BasicValues>
</OverdriveEffect>
</BlockBehavior>