EDGE Reference Page

This reference page exists as a condensed and continually updated primer on EDGE metrics. Although EDGE specifically refers to just one of the several metrics, the concept of **E**fficiency-**D**erived **G**oal **E**quivalents is applicable to all of them, and therefore we use "EDGE metrics" generically to refer to all of the Better Box Score Measures.

What is EDGE?

EDGE is a family of player metrics that combine familiar box-score statistics into summary measures of performance, efficiency, and impact, and which seek to avoid the shortcomings of existing metrics such as plus/minus to improve comparability across players, games, and leagues.

The general approach

Sports metrics divide into roughly two groups: those based on "box score" counting stats and those based on game-state data from each "event" within the game.

The outstanding Shown Space website is a terrific example of the latter, with a suite of metrics built from UFA game-state events, including metrics that can *only* be produced at this "each throw" level of detail, such as Win-Probability Added and Lag Contribution (a "measure of how much a player's passes improve their teammates' next throw").[1]

BBSM, as its name makes clear, is the former. The premise of BBSM is that, because ultimate is structurally a simple game, a well-designed and calibrated box-score model should be able to capture much of the same meaningful signal in overall player performance that an event model does. (At least for offense; as in most sports, we have insufficient ways of "counting" defense.) The advantage is that a box-score model is far less data intensive, and generally speaking, easier to understand: here are the 6 inputs and how each is weighted. Even if one is tracking all throws through programs such as Statto, converting those inputs to robust outputs is far easier with a box-score model.

But ease of calculation can't be the only standard. Easy is why we have the ubiquitous but highly misleading +/- metric. Although the action in ultimate is structurally simple, one aspect of ultimate—relative turnover risk, including the profound effects of wind—introduces significant distortions between players and games that confound one-size-fits-all box-score models such as +/-.[2] These elements are part of what we refer to as the game "environment," and the EDGE framework is designed to allocate "goal equivalents" among players for any given game environment, consistent with the number of goals scored in that game. The thrust of BBSM is to emphasize how to use basic stats to best effect.

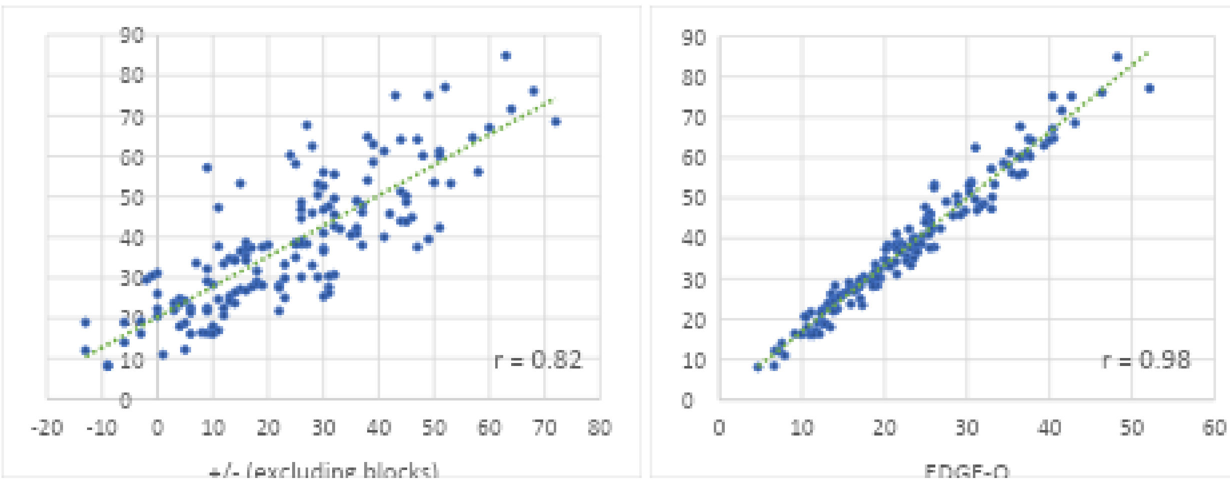
---

[1] The Western Ultimate League's excellent stats portal also contains several event-based metrics such as Expected Completion Percentage.

[2] The problems with plus/minus can be summed up in this fact: the 2025 WUL Offensive Player of the Year Ari Nelson had a season plus/minus of -0.5, tied for 137th out of 195 players. How? Because plus/minus overweighs the value of turns relative to scores in the WUL and then intensifies the effect for dominant throwers like Nelson. WUL voters wisely ignore plus/minus in favor of less distortive metrics.

If we accept that the granularity and statistical rigor of Shown Space metrics, such as its offense-focused Total Adjusted Expected Contribution (Tot-aEC), are the closest current method for identifying the theoretically "real" performance signal, Shown Space's existence helps validate both of BBSM's propositions: 1) a good box-score model such as EDGE can also reliably capture performance differences, and 2) traditional +/- is too simplistic to do so (Figure 1).

Figure 1. Correlation of Tot-aEC (the y-axis in both graphs) with +/- (left graph) and with EDGE-O (right graph). Data are 2025 totals for all UFA players with at least 200 possessions during the 2025 season. Because Tot-aEC is an offense-only measure, here +/- is calculated as Goals + Assists minus Turnovers. Pearson correlation r values shown in lower right corner of each graph.



Main components of the EDGE approach

A key reason EDGE-O correlates more strongly with Tot-aEC than +/- is the incorporation of summary yardage data. Yardage is a particularly important addition to the stat line-up, for several reasons:

1. Ultimate is a field-position-based game in which [scoring chances increase as you gain yardage] toward the goal, and therefore yardage is directly related to scoring and winning.
2. Yardage occurs on every throw, rather than being a sampling of activity like goals and assists. Because of this, and because there is so much yardage—thousands of units in every game—its distribution among players is less random than the distribution of goals and assists, particularly in single-game settings.
3. Given that it integrates so much activity in a game, yardage is a more robust proxy than scores for other qualitative skills such as breaking a mark, getting a disc off the sideline, consistently getting open, or making difficult catches. While true that not every 10-yard throw has the same impact, EDGE assumes that impact and total yardage for any given player largely align over the course of a game.[3]

Turns, turns, turns

[3] This assumption has a basis. EDGE started as a simple event-based and spatial model in which field-position value decreased as you moved toward the sidelines, with the intent of identifying players particularly good at getting the disc to more advantageous field positions. When we found that any player differences in a game were just as well explained by vertical yardage, we went in the direction of simplifying the approach. But more sophisticated game-state models like Shown Space are built for challenging the assumptions of summary stats.

Despite the many virtues of yardage, the starting point for EDGE is turns. From a "goal equivalent" perspective, the value of turns depends on their scarcity. In a game with only 10, the likelihood of getting the disc back after a turn is much lower than in a game with 60. EDGE reflects game-specific conditions by setting the value of turns equal to the game scoring efficiency (Goals/Goals + Turns), and from that starting point calculates the other factors. Because wind is so influential, EDGE generally calculates turn values at the game level so that season totals are correctly weighted.

What about defense?

When the only defensive activity we record is blocks, there is a choice to make. If we just use the "turnover value" of blocks, the amount of available defensive goal equivalents is much less than what is available on offense, and total "offense + defense" values will skew to offense (as in most sports). The second option is to treat blocks as a proxy for all defensive activity, but we don't think there is a good case for that, and given the relative infrequency of blocks, extrapolating their value would likely produce more noise than signal. We therefore stick with the first option.

Qincs

This is a UFA-only issue. Because turns are both costly and rare, counting them accurately is important. The UFA's timed quarters introduce an element to the game that does not exist in other formats: the desperation throw as time runs out. BBSM calls these Qincs—Quarter-ending Incompletions, and they should not count against a thrower's turnover total. Real throwing errors are failures of judgement, skill, or both, but Qincs are the opposite. By counting them as turnovers, the UFA unfairly inflates the turnover rate of potent throwers. (In 2026, this practice has eased slightly, but is now highly inconsistent, with about 3 out 4 instances still being recorded as a turnover.). Therefore, EDGE applies a correction, removing any throwing error from a player's total that occurs as the last event of a quarter. But it's not risk-free; if the throw leaves enough time on the clock for the opponent to make even just one more throw, it's a turnover.

Dealing with small sample-sizes

Small sample sizes can make *rates* highly misleading (e.g. 2-for-2 on pass attempts is a higher percentage than 99-for-100), which is why sports typically invoke minimum thresholds in order to qualify for certain metrics. But robust thresholds in ultimate can leave much of the league excluded, given how few games are played in a tournament or even a season, particularly for D-line players.

A more inclusive alternative to thresholds is to use empirical Bayesian estimation "priors," seeding each player a set number of events (touches, possessions, etc.) at the league average rate of what's being measured. In doing so, we assume players are average until they produce enough evidence to suggest otherwise; as sample size increases, the influence of the prior wanes. Bayesian smoothing makes our rate calculations more mathy, but the extra steps are worth it.

To sum up, BBSM:

1. Applies a framework of "goal equivalents" to weight different input factors and assign values.

2. Addresses significant sources of bias and risk, such as wind or the turnover risk differences between handlers and cutters.
3. Does not assume that team stats have the same meaning when applied at the player level. For example, why does "completion percentage" at the player level include "drops" by a teammate? Only because that is how it is calculated at the team level.
4. Employs several stats techniques to improve comparability, including empirical Bayesian estimation and standardization.

With that as background and rationale, here is a more detailed look at the main BBSM stats, which are oriented toward three main concepts: production, productivity, and efficiency. For simplicity, we will refer to handlers as those players who throw for more yards than they catch, and cutters as the opposite.

The Suite of EDGE Metrics

We will start with one table showing all the metrics and their relationship to each other. The methodology for deriving goal equivalents (GE) for yardage (Y_{ge}), scores (S_{ge}), and turnovers (T_{ge}), as well as empirical Bayesian estimation values represented by α and β, are described further below.

Table 1. The suite of BBSM EDGE metrics.

	Production	
EDGE-O	$Y_{ge} + S_{ge} - T_{ge}$	Offensive goal equivalents (GE)
EDGE-B	B_{ge}	Block GE
EDGE	EDGE-O + EDGE-B	Combined production, in GE
EDGE-PM	$G_{ge} + A_{ge} + B_{ge} - T_{ge}$	EDGE-adjusted plus/minus, when yardage not recorded
	Productivity	
xEO	100 * (EDGE-O+ α) / (xOO + (α + β))	EDGE-O per unit of expected EDGE-O opportunity
xEB	100 * (EDGE-B + α)/ (xDO + (α + β))	EDGE-B per unit of expected EDGE-B opportunity
xE	xEO + xEB	Combined productivity
	Efficiency	
CP+	$(Tch^{+} - AdjT + \alpha)/(Tch^{+} + (\alpha + \beta))$	Completion Percentage, modified
PE-O	$(THR_{ge} + REC_{ge} + \alpha) \div (THR_{ge} + REC_{ge} + AdjT + (\alpha + \beta))$	Player Offensive Efficiency
PE-B	(AdjB + α) ÷ (OppPoss + (α + β))	Block rate, SE-adjusted
PE	PE-O + PE-B	Player Efficiency
PER-O	100 * (PE-O/LgAvgPE-O)	Player Offensive Efficiency Rating
PER	100 * (PE/LgAvgPE)	Player Efficiency Rating
	Composite	
CR	Blend of 2-3 of the above	Composite Ratings, e.g. GameRating

EDGE (Efficiency-Derived Goal Equivalents)

The metric actually called EDGE is an allocation of <u>production</u> using Yards (Y), Scores (S), Turns (T) and Blocks (B) measured in "goal equivalents" (ge). Offensive production (EDGE-O) is calculated separately from block production (EDGE-B), then combined for EDGE. We'll begin with EDGE-O.

EDGE-O

EDGE-O allocates "goal equivalents" based on players' shares of the production that produced the team's goals, such that the sum of player EDGE-O values on a team approximates the team's goal total. EDGE is a simple additive function that subtracts the cost of turnovers from the value of generating yards and scores.

EDGE-O = (Y * Y_coef) + (S * S_coef) – T * GmSE

To get the yardage and scoring coefficients, at the single-game level:

1. Set the turnover value for both teams equal to the game's scoring efficiency (GmSE), which is the game's average scoring probability per possession.
2. Create a pool of "gross" total goal equivalents by adding turnover goal-equivalents (Tge) to the total game score. GEgross represents the total amount of production needed to achieve the goal total despite the negative effect of turns. This pool of goal equivalents will be allocated across all the players in the game based on their shares of production and turnovers.
3. Split the GEgross pool into yardage production (75%) and scoring production (25%), and calculate coefficients for each. Why the split? Yardage and scoring are fairly well correlated, but because 1) the last yard is by far the most important yard, 2) some players score disproportionately to their yardage, and 3) we already have scoring data, it's worth using both. We allocate 75 percent to yardage and 25 percent to scoring, roughly representing the relative value of the last yard.

Y_coeff = (Goals + (Turns * GmSE) * 0.75)) / Total yards [thr + rec]
S_coeff = (Goals + (Turns * GmSE) * 0.25)) / Total scores [goals + assists]

However, particularly for pro leagues, the scope of comparison is typically the whole league, so instead of calculating game-specific coefficients, we use league-average coefficients calculated from league totals of scores and turns.

Y_lg_coeff = (LgGoals + ((LgT * LgSE) * 0.75) / Lg yards [thr +rec]
S_lg_coeff = (LgGoals + ((LgT * LgSE) * 0.25) / Total scores [goals + assists]

This way, 100 yards and 2 scores for Player A in a game will have the same value as 100 yards and 2 scores for Player B in a different game. It is therefore more precise to consider these production GE values as "expected" GE, and not a strict allocation of Gross GE.

By incorporating yardage, EDGE-O addresses the risk imbalance borne by handlers. Across the pro leagues, approximately 40 percent of players throw for more yards than they catch. Therefore, the pool of throwing yards is divided up among fewer players than the equal-sized pool of receiving yards, such that the extra available Yge for handlers tends to balance out the extra risk. In traditional +/-, handlers are underrepresented among league leaders because there is no mechanism for balancing the disproportionate throwing risk; indeed, cutters take a larger share of assists than handlers take of goals, exacerbating the imbalance.

Example

The following table works through the EDGE-O steps for a hypothetical scenario in which Jojo was a key contributor to Team A's 15-13 win over Team B.

	Team A	Team B	Jojo
Goals	15	13	1
Assists	15	13	3
Turns	20	22	2
ThrY	1300	1100	200
RecY	1300	1100	100

Steps	**Formula**	**Example**
1. GmSE	GmG / (GmG + GmT)	(15 + 13) / (15 + 13 + 20 + 22) = 0.40
2. GEgross	GmG + (GmT × GmSE)	(15 + 13) + (42 × 0.40) = 44.8
3. Y_pool	0.75 × GEgross	0.75 × 44.8 = 33.6
4. S_pool	0.25 × GEgross	0.25 × 44.8 = 11.2
5. Y_coeff	Y_pool/(ThrY + RecY)	33.6 / (2400+2400) = 0.007
6. S_coeff	S_pool / Gm(G + A)	11.2/ 56 = 0.2
7. Yge	(ThrY +RecY) × Y_coeff	300 × 0.007 = 2.1
8. Sge	(G+A) * S_coeff	(3+1) * 0.2 = 0.80
9. Tge	T × GmSE	2 × 0.40 = 0.80
10. EDGE-O	Yge + Sge − Tge	2.1 + 0.8 − 0.8 = 2.1

Jojo's 2.1 goal equivalents represent 14 percent of Team A's 15 goals, a very substantial contribution.

Game-Level Terms

GmG — Total goals scored by both teams
GmT — Total turnovers by both teams
GmSE — Game scoring efficiency (scoring probability per possession)
GEgross — Gross goal equivalents (pre-turnover deduction pool)
Y_pool — Yardage goal-equivalent pool (75% of GE_gross)
S_pool — Scoring goal-equivalent pool (25% of GE_gross)

Player-Level Terms

ThrY — Throwing yards
RecY — Receiving yards
G — Goals
A — Assists
T — Turnovers
Yge — Yardage goal equivalents
Sge — Scoring goal equivalents
Tge — Turnover goal equivalents
EDGE-O — Efficiency-derived goal equivalents: offense; or "net offensive goal equivalents"

EDGE-B
The value of a block (to the defense) is set equal to the cost of the turn (by the offense). That is, blocks are also equal to the GmSE. In the Jojo example from above, each block would be worth 0.40 ge.

EDGE-B (or Bge) = B * GmSE

EDGE
EDGE-O + EDGE-B, the sum of total goal equivalents produced. Because EDGE-O is already calibrated to approximate team goals, team totals of EDGE do not equate to a particular game outcome.

xE (EDGE per "expected EDGE opportunity")
Per-game statistics are an incomplete basis for comparing players in ultimate because of disparities in opportunities within a game. Per-point would be a much better basis of comparison except for one thing—the differences in GE-generating opportunities between O-lines and D-lines.

For many years, BBSM used "possessions" as the rate-unit measure of choice. Offensive output could be set against the number of possessions played from both O-line or D-line, while a block rate could be set against the number of opponent possessions. However, just as wind increases the rate of turnovers, so does it escalate the number of possessions, and dividing any value by the number of possessions will systematically penalize players who happen to play in high-turnover conditions. A player who typically has 20 possessions in a game but accumulates 40 possessions in a single wind-affected game has not had two games' worth of offensive opportunity; they have had one game's worth of opportunity fragmented across many lower value possessions.[4] This is particularly the case in BBSM's goal-equivalent (GE) framework, where turnover values adjust to their scarcity; if possession values don't likewise adjust, per-possession rates will skew.

One fix would be to adjust the observed possession-count based on the league-average, but we now favor an approach that mirrors our treatment of turnovers, produces output that is more intuitively interpretable, and enhances comparability across leagues: multiply possessions by the GmSE. This creates new denominators for our "rate" metrics, now built around "opportunity" rather than static possession counts: expected offensive opportunity (xOO) and expected defensive opportunity (xDO). Using xOO and xDO allows us to answer this question: of the goal-equivalent opportunity that existed for your lines, how much did you personally realize?[5]

As an extra but not required step, we also use empirical Bayesian estimation with xE so that we can evaluate productivity rates throughout the season (including for single games) and not create

[4] Extreme events such as Michigan's 8-7 win over Brown in a 2026 Florida Warm-Up quarterfinal can bring certain methodological shortcomings into sharp relief; played in sustained 30-40 mph winds, the teams combined for an estimated 136 turnovers, resulting in 151 possessions. Per-possession rates make no sense in this context.
[5] To be precise, in a summary stats model such as this that uses the "average" possession scoring rate of the game, we are talking about probability-weighted opportunity.

arbitrary thresholds.[6] In doing so, we seed each player a given number of observed possessions (for xEO) or expected opponent possessions (for xEB) at the league-average xEO or xEB rate. (More info on Bayesian priors is covered below in PER.) Based on analysis of full-season data, we have set the prior possessions at 30, a modest level that works well throughout the season. For more detail, we'll look at xEO and xEB separately.

xEO and xEB

xEO is EDGE-O per expected offensive opportunity (xOO), expressed as a percent; xEB is EDGE-B per expected defensive opportunity (xDO). Combined with our Bayesian estimation priors, the equations are:

xEO = 100 * (EDGE-O + 30 * LgAvgEDGE-O) / (xOO + 30)
xEB = 100 * (EDGE-B + 30 * LgAvgEDGE-B) / (xDO + 30)
xE = xEO + xEB

where:
xOO = observed possessions * GmSE
xDO = observed opponent possessions * GmSE

Removing the Bayesian components for the sake of a simple example, if Jojo had an EDGE-O of 2.0 in a game in which they played 20 possessions, and the game itself had a GmSE of 0.40:

xEO = 100* 2 / (20 * 0.4) = 200 / 8 = 25

The advantage of xEO is that we can quickly interpret this score. xOO represents the total expected offensive opportunity for the entire 7-player line on each possession, and each individual player's EDGE-O is their personal capture of that shared opportunity. In club play, this means that the average xEO is 100/7, or 14.3. (Because of player substitutions during time-outs in semi-pro leagues, the average is lower; in the UFA, it's about 12.7). Jojo's 25 xEO is almost twice the average, indicating a highly productive and impactful performance.

We should note that xEO, like its predecessor EO100, conflates two factors: 1) the player's involvement and efficiency and 2) the line's efficiency. A line that converts opportunities at a higher rate than average generates more GE per xOO unit for everyone on it, and a player on a strong offensive line benefits from this even if their individual contribution is modest. In our view, to the extent that high xEO percentages have a small bias towards winning, fine. Nevertheless, we have introduced Player Efficiency Ratings (PER) below to attempt to disentangle these two dynamics. We have also retired E100.

<u>When possession data is not available</u>

The PUL currently does not publish player possession data, so for xE we estimate possessions based on GmSE and O-line/D-line points played splits:

xPoss = (OLP * (1/(2-GmSE) * 1/GmSE) + (DLP * (1-GmSE)/(2-GmSE) * 1/GmSE)
xOpPoss = (OLP * (1-GmSE)/(2-GmSE) * 1/GmSE) + (DLP * (1/(2-GmSE) * 1/GmSE)

where:
OLP = O-Line Points Played
DLP = D-Line Points Played

---

[6] In recent years, we used an E16 metric to address the issue of "possession" sample-size, but applying priors to xE has more advantages, including not having overlapping metrics, so we have removed E16 from the line-up.

EDGE-PM (When all you have is G/A/B/T)

Yardage is great, but outside of semi-pro, the Goals/Assists/Blocks/Turns stat line remains the template. So how can we combine these four inputs in a way that minimizes the deficiencies of applying equal weights like +/-? We answered that question here, and below is the table of adjusted coefficients for EDGE adjusted plus/minus, or EDGE-PM. Like EDGE, EDGE-PM begins with setting the coefficients for Turns and Blocks equal to the game scoring efficiency. For a tournament in which teams have played under generally the same weather conditions, one can also apply a tournament-level SE for the relevant division. Division SE's should be calculated separately. Once the SE is known, find the closest SE in the left column and apply the coefficients in that row.

SE	*G*	*A*	*B*	*T*
0.25	0.75	1.00	0.25	0.25
0.30	0.73	0.97	0.30	0.30
0.35	0.71	0.94	0.35	0.35
0.40	0.69	0.91	0.40	0.40
0.45	0.67	0.88	0.45	0.45
0.50	0.65	0.86	0.50	0.50
0.55	0.62	0.83	0.55	0.55
0.60	0.60	0.80	0.60	0.60
0.65	0.58	0.77	0.65	0.65
0.70	0.56	0.74	0.70	0.70
0.75	0.54	0.71	0.75	0.75
0.80	0.52	0.68	0.80	0.80

Efficiency

EDGE and xE are designed to sort out overall impact on the field, and on winning. For xE, one player's production must be balanced by another player's lack of production, since the amount of production is finite. Everyone cannot do everything on every point. While it's therefore not possible for every player to maximize xE, it should be possible for every player to play with maximal efficiency; that is, create production with no waste (i.e. no turns).

Production and productivity also strongly reflect team and line dynamics, over which any one player may have little influence. Even if your own performance is flawless, if your line frequently turns the disc over before anyone has a chance to generate production, everyone's per-possession rate is going to suffer. This is the basis for our distinction between productivity and efficiency: efficiency implies a ratio of output to input where the same agent controls both.

We therefore look to efficiency stats to help isolate individual performance from team dynamics, although it's hard to do so completely—it's far easier to complete passes if you have superior cutters and reset handlers who are always open, or if your teammates took huck attempts early in the game that has opened up the unders. But if we better isolate the things a player more directly controls—namely their own turnovers, not the team's—we may better illuminate those who are playing well on weaker teams or lines, more fully credit excellent players on very balanced teams, highlight those who are both incredibly productive *and* efficient, or simply underscore the value of being rock-solid, even if not prolific.

Two efficiency stats already widely in use are Completion Percentage (CP) and Team Scoring Efficiency (SE). In each case, instead of counting turns as negative offsets as in EDGE, they are calculated as percentages with the basic construction of Good/(Good + Bad). We will riff on each of these, first offering an improvement to CP, and second using Team SE as a model for Player Efficiency (PE).

Completion Percentage Plus (CP+)

CP+ modifies the common Completion Percentage (CP or Comp%) metric, offering a more robust comparison of relative turnover propensities through the following differences:

1) CP+ is a comparison of overall disc retention, not just throwing incompletions, and is calculated on a "per-touch" basis instead of "per attempt."
2) Unlike CP, passes that are dropped by the receiver do not count against the thrower since they don't reflect on the thrower's accuracy.
3) Unlike CP, passes dropped by the would-be thrower <u>do</u> count; one's own drops directly reflects on one's disc retention. BBSM's calculation of Tch includes these two adjustments; we call it Tch+.
4) Turns are GmSE-adjusted to control for strong wind effects.
5) To smooth the effects of low-sample size, we seed each player with prior touches at the league-average CP+ rate. Unlike xE, CP+ is a rate bounded between 0-1, and we have the option of dynamically deriving the priors from the mean and variance of population rates, but to keep CP+ more straightforward, we will again fix the prior count at 30.

$$CP+ = 100 * ((Tch^{+} – AdjT + (30*LgAvgCP+)) / (Tch^{+} + 30))$$

In which:

Tch^{+} = Throws + Drops + (Goals * 0.5)

Goals are only counted as half a touch because there is no opportunity to complete the more difficult part of the touch, whereas drops are counted as a whole touch because the drop forecloses the chance of completing the ensuing pass.

AdjT = T * (GmSE/LgAvSE)

Players who commit more turns under windy conditions will have AdjT values smaller than their number of raw turns.

.

Player Efficiency (PE) and Player Efficiency Rating (PER)

CP+ is great for evaluating overall disc retention, but in valuing "lack of turnovers" above all else, CP+ minimizes the role of relative risk, and does not indicate who is generating the most *production* per turnover. For that, we developed Player Efficiency (PE) values and Player Efficiency Ratings (PER).[7]

[7] "Yards per Turn" is quick and easy, and many use it, but if suffers from the main issues identified here: very unstable at low turnover values, and bias against handlers. "Throwing Yards per Turn" flips the bias in favor of

Since team possessions equal (G+T), and scoring efficiency is therefore G/(G+T), if we take the EDGE approach of assigning coefficients to yards and scores to produce goal equivalents (GE), we can convert team scoring efficiency (SE) into player efficiency (PE): GE/(GE+T). Just as team efficiency is goals per possession, PE is player goal equivalents per "player possession."

For the team calculation, it doesn't matter who committed the turnover, and therefore the higher risk of throwing is irrelevant. At the player level, it's very relevant. So is the problem of small sample size. Before going through a detailed example, I'll highlight PER's approach to these two issues: risk weightings and empirical Bayes estimation.

The weighting is not the hardest part

In the "good minus bad" arithmetic relationship of EDGE, the additional risk of throwing is adequately compensated for by the extra share of yardage available to majority throwers. In efficiency calculations, there is no yardage advantage since the calculation is essentially on a per-yardage basis. Therefore, we turn to weightings.

Our method for determining weights is to calculate league throwing and receiving efficiencies separately, and then identify the weightings one would need to apply so that the two efficiencies combined equal overall league efficiency.

As an example, in the UFA 2025 season, there were 4613 adjusted throwing errors and 816 adjusted drops.[8] From this, one can calculate the league-average "throwing" efficiency and "receiving" efficiency, given that there were 3039 goal equivalents for each throwing and receiving (resulting in a league total of 6078 goals):

Overall league efficiency: 6078 / (6078+4613+816) = 0.528
League average throwing efficiency: 3039 / (3039+4613) = 0.397
League average receiving efficiency: 3039 / (3039+816) = 0.788

To get the weighting factors, we divide the league efficiency by the separate efficiencies:

Throwing: 0.528 / 0.397 = 1.33
Receiving: 0.528 / 0.788 = 0.67

These factors are applied to each player's goal-equivalent values for throwing (throwing yards and assists) and receiving (receiving yards and goals).

Empirical Bayesian estimation

As stated above, a more inclusive approach than cutting out all the players below a certain sample-size threshold is to use empirical Bayesian estimation, in which one estimates "priors"

handlers. Use with caution. PER essentially combines Throwing Yards per Turn and Receiving Yards per Turn into a composite, while including the value of scores, and adjusting for sample size.

[8] UFA reported non-adjusted totals were 5020 and 858; BBSM adjusts these totals by removing 304 Qincs from throwing errors and adjusting both throwing and receiving errors for the game environment (primarily wind effects).

based on the distribution of player values, and then use each player's actual data to update the prior. In effect, the technique assumes every player is average until enough data accumulates to indicate otherwise. Those with small sample sizes will see efficiency values pulled very strongly toward the mean; those with a robust resumé will see relatively less effect, and marginally less effect with each additional throw until the effect is imperceptible.

In the xE and CP+ examples above, we fixed the prior value across all leagues and all seasons, but because PE is more sensitive, we calculate α and β dynamically for the specific dataset, using the following beta-binomial equations.

α = (((1-(AVG)/VAR)-1/AVG)*AVG^2
β = α * ((1/AVG)-1)

<u>Mitigating waste through defense</u>
Our conception of efficiency is the ability to produce without waste, so what if a player, after every single one of their turnovers, generated a block on the ensuing possession—shouldn't we consider that the block "recycled" the waste, such that it was no longer waste? If we consider blocks as an efficiency offset, the calculation question is fairly straightforward: if every turn is a loss of a possession, then what is the likelihood that the player will regain the possession through a block?

That answer is the number (not goal-equivalent value) of blocks per opponent possession (adjusted for GmSE). After an empirical Bayesian adjustment for this percentage as well, we can just add the AdjB values to the PE.

Because the offense-only PE is a useful value itself, we will rename it PE-O to set it apart from the broader-scoped, blocks-included PE.

<u>One last step: converting to PER</u>
Player PE values can be interpreted in the context of league-average scoring efficiency; for example, in the UFA, a PE of 0.60 would be well above the league average SE of 0.52. The simpler way to convey this, as well as to compare players across leagues, is to represent the scores as a percentage of the league average, giving us Player Efficiency Ratings (PER) and Player Efficiency Ratings-Offense (PER-O), with the league-average player = 100.

A PER example
The 2025 seasons of Allysha Dixon of Alpenglow and Abbi Shilts of Super Bloom serve as good, high-efficiency examples.

Table 2. 2025 season counting stats for Abbi Shilts and Allysha Dixon. Adjusted Turns (AdjT) and Adjusted Blocks (AdjB) are BBSM adjustments to account for game-specific conditions, primarily wind.

Name	Poss	OpPoss	Tch	G	A	B	Thr Y	RecY	Y	EY	T[9]	AdjT	Adj B
Abbi Shilts	85	81	62	8	9	7	412	594	1006	1099	3.5	3.0	7.0

[9] The WUL splits tough "throwaway vs drop" calls, and assigns 0.5 to both thrower and receivers. Seems fairer.

Allysha Dixon	134	122	120	8	9	2	869	499	1368	1463	4	4.1	2.1

Step 1: Adjust turn and block totals

Adjust player turn and block totals at the game level to help control for game conditions by multiplying them by an SE factor (GmSE/League Average SE). Note that the adjustment lowers Shilts's turns by 15 percent, indicating that her throwaways disproportionately occurred in low-SE games.

Step 2: Calculate ThrGE and RecGE factors

Per the UFA example above, calculate the ThrGE_factor and the RecGE factor, based on league-average throwing and receiving efficiencies. However, for the WUL, instead of using total yards, we use their Effective Yards (EY) stat, which helps contextualizes throwing errors by adding the distance of incomplete throws to yardage totals, thereby slightly favoring those who average longer throwaways. Because EY adds more yards to throwing, ThrGE and RecGE are not equal, and the ThrFactor and RecFactor must be calculated as:

ThrFactor = 2* (ThrGE+AdjThrErr)/(ThrGE+RecGE+AdjThrErr+AdjRecErr)
RecFactor = (2-ThrFactor)

In the end, the WUL factors below are similar to the UFA's 1.33 and 0.67.

LgThrEff	LgRecEff	TotalEff	ThrFactor	RecFactor
0.285	0.523	0.356	1.40	0.60

Step 3: Convert raw stats into goal equivalents

As with EDGE, we do this by generating yardage and scoring coefficients by dividing 75 percent of the league total of goals by total yardage (throwing + receiving), and the remaining 25 percent of goals by total scores.[10] For the WUL, we use Effective *Throw* Yards (EThrY), which is EY-RecY.

PE_GE = ThrGE*ThrFactor + RecGE*RecFactor

Where:

ThrGE = (EThrY * Y_coef) + (Ast * Scr_coef)
RecGE = (RecY * Y_coef) + (Goals * Scr_coef)

Shown below, as a dominant thrower, Dixon's PE_GE is increased 10 percent by the weightings, while Shilts's remains about the same given the balance of throwing and receiving.

	ThrGE		ThrFac		RecGE		RecFac		PE_GE	PE_GE_unweighted
Shilts	2.81	*	1.4	+	2.98	*	0.6	=	5.72	5.79
Dixon	4.34	*	1.4	+	2.66	*	0.6	=	7.68	7.00

Step 4: Calculate raw PE-O values

[10] Unlike EDGE, since this is not an additive function in which we will subtract out turnover values, we do not first calculate a "gross" pool of GE.

PE-O_raw = PE_GE/(PE_GE + AdjT). PE-O indicates that we're just calculating the offensive efficiency at this point.

Shilts 5.72 / (5.72 + 3.0) = 0.66
Dixon 7.68 / (7.68 + 4.1) = 0.65

Step 5: Adjust for sample size

Is Shilts's 62 touches and Dixon's 120 touches enough information to deduce that they are each far above the league average in offensive efficiency? The empirical Bayesian estimates based on the mean and variance of the raw 2025 WUL PE-O scores resulted in adding 10.76 personal possessions to the denominator, which at league efficiency rates means adding 3.93 GE to the numerator. Because Dixon had more overall activity, her score shrinks less, and although trivial in the context that both were outstanding in 2025, their PE-O ordering swaps.

Shilts (5.72 + 3.93) / (5.72 + 3.0 + 10.76) = 0.495
Dixon (7.68 + 3.93) / (7.68 + 4.1 + 10.76) = 0.515

Step 6: Calculate PE-B

Calculate the value of getting blocks in this efficiency framework (PE-B), using a separate Bayesian prior count ($\alpha + \beta$) adjustment to control for sample size.

$$PE\text{-}B = (AdjB + \alpha) / (OpPoss + (\alpha + \beta))$$

Where, with a league-average block rate of 0.033:

$\alpha = 1.52$
$\beta = 41.1$

Given that Shilts had 50 percent fewer OppPoss than Dixon, the Bayesian shrinkage moves Shilts more toward the mean than Dixon, although Shilts's new value is still nearly twice the league average.

	AdjB	OpPoss	AdjB_rate_raw	PE-B
Shilts	7.0	81	0.086	0.064
Dixon	2.1	122	0.017	0.021

Step 7: Combine PE-O and PE-B

Because both metrics are already on a "per-possession" rate basis, we can simply add PE-B to PE-O for an overall player efficiency score.

Shilts: 0.495 + 0.064 = 0.559
Dixon: 0.515 + 0.017 = 0.532

Step 8: Convert to Ratings

Given that the league average PE-O and PE were 0.361 and 0.394 respectively, it's clear that both Shilts and Dixon were very efficient, but this is more clearly conveyed by representing the scores as a percentage of the league average, with the league-average player = 100.

In our examples:

Shilts PER-O = 100 * 0.495 / 0.361 = 137
Dixon PER-O = 100 * 0.515 / 0.361 = 143

Shilts PER = 100 * 0.559 / 0.394 = 142
Dixon PER = 100 * 0.532 / 0.394 = 135

Dixon led the league in offensive efficiency (PER-O), while Shilts led the league in total efficiency (PER). Both players are good examples of highly effective players who in 2025 shared possessions with so many other highly effective players (and a few dominant players) that their share of total production was suppressed. PER offers that additional perspective.

Composite Ratings (CR)

The three pillars of production, productivity, and efficiency emphasize different and valuable aspects of a player's performance. So when it comes to MVP, All-Team selections, and the like, when we're looking for one value to represent the quantitative case, BBSM favors a blend rather than a single-origin bean. CR is the broad term for this blend, but it takes specific forms such as GameRater (GR) and SeasonRater (SR).

What is blended depends on the objective. For example, while the MVP-oriented SR might include EDGE, xE, and PER, Offensive Player of the Year would focus on EDGE-O, xEO, and PER-O. Our Break Generators metric combines EDGE-B with xEB. We do not have fixed proportions for the composites, particularly as we have just added PER, and to some extent, the proportions depend on the context. For example, for MVP considerations we tend to heavily weight *production* to better capture the notion of "what did you do this season?" But a forward-looking "best player" discussion would lean more heavily on the rate metrics. We use z-scores to combine metrics of different scales.

Everything we do in BBSM is reductionist, but CR takes it to another degree; while such metrics are intriguing for ranking top players and performances, we don't see value in applying them at the other end of the performance scale. Therefore, we use CR on our leaderboards but not include in full-league tables.

-- END --