

What is your work schedule?

After the first 90 Sols we were following Mars Time, we now have a regular work schedule, typically 8:00 to 20:00 Pacific Standard Time. We work all days of the week, including weekends, we did not work on Thanksgiving weekend (end of November) or Christmas. For those holidays we prepared the commands for several days in one batch. So we got to stay home but the rover had to work during all the holidays. We will transition to 6 days/week operations and probably after Solar Conjunction we will transition to 5 days/week operations. The rover will still work every day, she does not get any days off :-)

How do you test your commands for next day driving?

How many people check commands before sending?

Do you use emulator to test your commands before loading onto the rover?

What do you do if find a mistake in sent commands?

These are excellent questions! We prepare the commands using software that has been developed by JPL for this purpose. We use a specialized editor, called RoSE (Robot Sequence Editor) that checks basic mistakes like typos in the command name, the range of values for command parameters, etc. Connected to the editor we have a simulator called Hyperdrive. The simulator takes the images that are captured during previous Sols and shows the terrain in 3D. The simulator takes the list of commands and shows what the rover will do and how it will interact with the terrain. The simulator can also display basic simulated telemetry such as position and attitude but also what the flight software will say about the drive including some error conditions.

For critical activities such as robotic arm drilling we sometimes go and test the activities in the testbed prior to send them to Mars. Once we have a set of commands that we believe are correct, they are reviewed by other fellow drivers. Typically there are at least three drivers on each shift (one expert in driving, one expert in using the robotic arm and one expert in using the turret instruments) but sometimes more drivers stop by and look at the commands. During our shifts we have five formal reviews of the activities by a group of about a dozen people, not all rover drivers as the daily activities include all the various instruments and we need to make sure there are no bad interactions that could cause faults or even worse, damage to the vehicle. We have other software tools that cross-check our commands and a long checklist of things to verify.

We try not to make mistakes but they do happen and a big part of the software we have on the rover checks that all the commands make sense, that we do not try to make silly things like firing the laser at some part of the rover. So if we do not catch a mistake on Earth, the rover on Mars will find it.

How do you run tests here on Earth (do you have a copy of the rover somewhere in the desert?),

We have two vehicles. One is called VSTB (vehicle surface testbed) that has all the parts including cameras and robotic arm and we can command it exactly like the rover on Mars. Whenever we need high-fidelity testing this is the vehicle we use.

This is a picture: <http://photojournal.jpl.nasa.gov/catalog/PIA15876> The other vehicle we have is called "Scarecrow" and has only the mobility part (wheels and motors) that we use to test driving on various terrains.

We have two facilities at JPL, one indoors called In-Situ Instrument Laboratory

(<http://spacedoutclass.files.wordpress.com/2012/08/isil.jpg><http://farm5.staticflickr.com/4054/4646918337>

[2d227be960_o.jpg](#)). We also have an outdoors facility called the Mars Yard (http://farm4.staticflickr.com/3045/2548636741_c1ed310947_o.jpg) where we can do more driving tests. We have taken Scarecrow in the desert near Death Valley in California in May 2012 to run some testing driving on sand dunes prior to landing as we were concerned about the dark sand dunes Curiosity could have landed on.
<http://mars.jpl.nasa.gov/msl/news/whatsnew/index.cfm?FuseAction=ShowNews&NewsID=1222>

and how do the programmers learn and train?

we train each other. Some of us have experience in driving MER, some of us have built the rover, some have written the software we use to command the rover. It takes a lot of practice!

Are there any plans to provide the resources and time of Curiosity available to students, for learning science experiments, after the rover finishes all planned programs of research (if at that time it will be fully functional)?

There are a lot of scientists and many university students that are working on this mission whose job is to define and carry out experiments. I don't think we will ever finish a list of experiments as the list continually grows. Whenever we find the answer to one of our questions, 10 or more new questions arise!

Is there any plans from NASA to visit rover landing sites of the previous Mars devices (rovers/station)? Most of them have gone out, but it would be interesting to establish the true cause of failure in real place. By the way, photos of these once-legendary vehicles would have a very high aesthetic, artistic and retail value.

Currently the goal is to explore as much of the planet as we can so it would be less useful to return to previously visited places even if we returned with more science instruments. Understanding the cause of failure of our robots is less interesting than gathering new science data. So while I share your desire to see these rovers again, I don't think this will happen in the near future.

How is the 7 minute delay handled? Is it a major problem?

The distance from Mars varies from a minimum of 5 light minutes to a max of about 20 light minutes one way. We are currently close to the max (19 minutes). The delay is sufficiently large that we cannot control the vehicle interactively. Think about it, even at the closest distance when you see an obstacle in front of the rover and you send a stop command, the rover would receive the stop command 10 minutes too late! We control the vehicle by sending a pre-programmed sequence of commands that gets sent every day. We could not command the rover interactively even because the Earth is not always visible from Gale Crater where Curiosity currently is. And even if it was, a long-distance phone call to the rover is quite expensive, about \$10000 per hour!

The delay to receive data from Mars is usually much longer, though, as we relay the data through Mars

Odyssey and Mars Reconnaissance Orbiter. Typically it takes no less than a few hours to receive the data from the rovers.

Is there some piece of equipment missing on the Curiosity that you need badly?

More science instruments, a more powerful computer, more data storage, more downlink capability, more power, higher resolution cameras, sometimes a higher mast...

Were there occasions when you found non-critical bugs in the code during its execution at the rover?
How do you handle them?

Sure! On MER we discovered software bugs years after landing. One critical bug was found by MER Spirit on sol 18 but we found one in 2007 while driving around Victoria Crater. We have a team of people (including myself) that analyzes the telemetry from the rover and whenever we find something that does not seem right we go in the testbed and try to replicate the conditions and see if we can replicate the problem. Once we find the cause we can either modify one of the thousands of flight software parameters or we send a "patch", that is a small piece of software that gets inserted into the code that fixes the problem. Obviously, while the bug gets fixed we avoid using these commands or conditions.

What's the most challenging part of your duties?

Besides all the paperwork involved in driving Curiosity ;-), the most difficult part is imagining all the possible bad outcomes. While we have lots of experience in driving and operating a rover on Mars we always have to be on the lookout for something that could go wrong. While we are driving we always try to have at least one possible escape route for example. We also avoid activities that although they have a very low probability could have a catastrophic outcome.

Could you share with us some amusing incidents you've encountered?

One of the most amusing incidents happened in 2006 on MER. This mission was designed to last only 90 Sols and when the flight software was designed and written the Sol designation could hold only three digits. As Sol 999 was approaching we updated and loaded the software on board the vehicles. This was a monster task, it took one full year to complete the update and testing and several months to load the software to the rovers. The new flight software also needed new ground software as it was incompatible with the old one. As a result we needed to boot the two rovers on the new software on the same Sol as we could not have one rover running the new software and one on the old. That day finally comes. Spirit gets to reboot first, we send the command to boot with the new software, the rover sends back data, all is well. Opportunity is on the other side of the planet and we have to wait 12 hours before we send the command for her. Before that happens, the JPL ethernet backbone goes down and we cannot copy the commands from our server to the DSN station. Time is running out, the JPL backbone is not coming back! What do we do? Fortunately one of the computers in mission control has a direct link to the DSN station (Canberra I think) and that computer has a floppy drive. Fortunately we still had one of the MER computers that had a floppy drive, but who has a floppy disk in 2006? Run around the lab trying to find a floppy disk. We found one in a forgotten drawer, reformat the disk, copy the commands over it and run to mission control and

send the commands. We even had two minutes to spare!

How many people control the rover?

It depends on how you define "control". There are currently 16 rover drivers, but the team of people who control the various devices (cameras, SAM, CheMin, ChemCam...) is close to one hundred I think.

Are the rovers fool proof?

The saying is "nothing is fool proof as fools are too ingenious".

Suppose you program a trajectory for the manipulator that would damage the rover itself once executed (e.g. knock off an antenna). Would the execution of such code be banned automatically during test phase or later, or is it necessary to manually double check for such errors?

The on-board software checks if any parts can collide with the rover so in this case the command would not be executed, an error would be generated and the rover would stop the arm activities. If a subsequent activity would rely upon the arm being in the commanded position, such as firing the laser, that activity would not be executed as well.

Even if the on-board software protects the vehicle, we still double (and triple, and quadruple) check that these things do not happen. We also check that these collisions cannot happen even if the Arm or other components are not exactly in the expected position. It is sometimes impossible to predict the exact conditions on Mars. While we have good images and 3D position of the Martian Surface, the data might have some measurement errors and we might interact with the surface at an unexpected position. We need to ensure the activities can work even with some large errors.

How does the communication with the rover works? What antennae/modems are used? What low-level protocols? How do the higher level protocols handle the presumably high BER?

We send the commands directly from Earth to the rover HGA (High Gain Antenna) using one of the DSN stations. The link is on the X band (7-8GHz). This link has all the detail you can imagine:
http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&ved=0CDYQFjAA&url=http%3A%2F%2Fdescanso.jpl.nasa.gov%2FDPSummary%2FDescanso14_MSL_Telecom.pdf&ei=rhUAUbWzOKKhigLbulGACA&usg=AFQjCNEFr6yltmK2TDAsniaYHaRFk7pvjA&bvm=bv.41248874.d.cGE

The data that the rover gathers during a day's activity gets sent back using the UHF antenna (the black cylinder to the right of the RTG) to one of the orbiters (MRO and ODY). The orbiters forward the data to the DSN stations (I think on X band again). Typical volume of data returned for MER was about 100 Mbits while for MSL is much larger, on average about 500 Mbits but we had a monster pass of 1200 Mbits once.

What are the the most noticeable differences and considerations with shooting images on Mars,

compared to the conditions on Earth? What kind of extra knowledge should mars-photographer have to create that images? For instance as there atmospheric structure is different does some colors dominate/degrade more than others compared to Earth conditions, the similar way as it happens shooting images underwater? Considering first of all the images that are close to what human eye see, not invisible spectrum.

I think this question would be better answered by mcaplinger but I will try to answer anyway.

Mars atmosphere is much thinner than Earth's and much more predictable. There is some dust in the atmosphere so you have some degradation at larger distances. Sometimes the atmosphere color looks like the Los Angeles basin polluted air. Mars, being a bit farther from the Sun has a bit less light but overall I don't think there are many differences. In fact I think shooting pictures on Earth is more difficult, there is much more variations in colors, in brightness and contrast than on Mars.

Was the task of filming itself harder than rovers usual daily routine?

The cameras that Malin Space Science Systems have developed can capture video at up to 15 Hz (frames per second), they have internal memory, so for us is not complicated at all. We just ask the MSSS people to provide the commands to capture a video and that's all. The potential issues are length of video, frequency of frames and making sure the video begins at the appropriate time and cameras are pointed at the right location or device. Again mcaplinger might provide more in depth answers to this question.

How is the emergency program interruption implemented?

There are various levels of emergency actions. If there is an error in the commands we send, the rover simply refuses to execute them. If there is an error while executing a command (such as the arm example above) the rover will set an error condition and simply refuses to use one or a set of devices as a consequence of the error. If there is a software bug, that is a software condition that should not be possible is reached, the rover will reboot and go into "safe mode". In that case the rover will execute a pre-programmed set of activities including sending the error messages and will await instructions from Earth on how to proceed. If the software "hangs" there are watchdog times that will externally reboot the vehicle and make it go into safe mode. We do not have a blue screen of death.

What settings do you track on rover in «real time» or close to it for emergency control?

Hopefully the answer above includes the answer to this question.

Have the rover's own security system to stop the execution of the program?

On MSL I think we had a spacecraft reset during cruise but I don't remember what was the reason. We had many resets on MER, including many due to radiation hitting the processor and memory. Depending on the reason behind the spacecraft reset these are handled as a nuisance (such as cosmic radiation) to a major anomaly (such as Spirit Sol 18 anomaly).

Is there an algorithms to recover from the emergency stops or the program only stops and the unit go in standby mode?

See above, it all depends on what happened. There are some errors that are not a major disruption to the activities and some that can be a major hassle. Spirit Sol 18 anomaly is of course in the second category.

Does Curiosity use DTN architecture in data transfer protocol, what are the minimum latency possible?

The minimum latency is 5 minutes, the max is 20 minutes but since we do not operate the vehicles in real time, the delay is not hindering the operations. We do not use DTN but some techniques are used to handle the long distance communications. We typically gather much more data in one Sol than we can transmit so we have to pick and chose which data to downlink each Sol. Each piece of data gets assigned a priority, higher priority data comes down first, lower priority comes down later (if at all). When we acquire data we assign a priority but we can later change its priority if we decide that the data importance is different. For example if a thumbnail shows the camera was mispointed, we lower or delete the full-resolution image. Vice-versa if a better exposed image is available, we will increase its priority. Moreover, data is broken up into packets, this is important for large volume data such as images. This is why some of the images

you see might have holes. Some of the packets might get lost in transmission or might have transmission errors. In case of errors we can also retransmit data that gets lost.

Can it happen that all the wheels of the rover get stuck in the sand so that it cannot move any more?

Yes. Curiosity cannot drive on slopes larger than 25 degrees on any kind of terrain. On loose sand it cannot climb more than 12 degrees.

Is there some way to get out of the trap then?

We try not to get into those situations in the first place but if we have to, for example because we want to reach a particular science target, we try to have at least one escape route. For example Opportunity when she tried to climb the base of Cape Verde (and failed to do so) we had two escape routes.

How precise are clocks used on rovers? Can they be adjusted, and if yes — how?

Just to give you an idea, Opportunity clock after nine years is about 20 minutes off. We typically do not adjust the clock because it can disrupt data bookkeeping but we can measure the clock drift. We tell the rover to send a "beep" at a specific time and for a specified length of time. We then listen with a DSN station and see when the beep is heard. Actually we determine when the beep *ends* as we might miss the beginning of the beep. Since we know how long the signal takes from Mars we can determine quite accurately what is the clock drift.

It is obvious that Martian calendar differs from its Terrestrial counterpart. Please tell us more about calendar use on rovers.

Mars rotational axis is tilted, similar to Earth's, so Mars has seasons. Mars takes about two Earth years to complete one orbit around the Sun so the seasons are about twice as long. Since Mars orbit is not perfectly circular the seasons are not exactly equal in length. A martian day is called a Sol and is about 39 minutes longer than Earth's day. We do not have Martian Months nor Martian weeks but we have Martian seconds, minutes and hours. Our Martian calendar simply has the Sol number, that is, the number of Sols since landing. We have some neologisms: Tosol (today), Yestersol (yesterday), Morrowsol (tomorrow), Soliday (holiday).

What improvements in the rover-control process would you like to see?

Shortly we should be able to enable autonomous navigation on Curiosity. This will allow to increase the distance for each drive. Currently AutoNav analyzes only the shape of the terrain where the rover needs to drive. We are in the process of developing algorithms that can determine the type of terrain (bedrock, sand, dune, cobble, crater...) so that the vehicle can better gauge where it is safe to drive.

Does NASA have any plans to make rovers that could check their own technical state, print the needed spare parts on an internal 3D-printer, and repair themselves?

Not that I know of. It would be quite difficult to send a 3D printer to Mars ;-). It is simpler to have redundant systems. Curiosity has two sets of many things: computer, cameras, sensors etc... We can switch between them in case one fails. While we have only one set of wheels, if one motor fails, we can set the wheel in "neutral" so we don't have to drag it as we did with Spirit.

How many new lines of code do you need for writing «one day» program for rover?

We use a custom language to program the rover but one day can have as little as 1000 lines of code. Some of the code is reused and we keep it on board the vehicle, some of it is new. Some of the more complex activities such as sample acquisition and handling can take many thousands lines of code. The code we send daily looks more like a script than code, contains activities for more than one Sol, typically for three Sols. The first Sol that has the desired activities and then what we call "run out" Sols with some basic science and housekeeping activities for the other Sols. This is done so that if we cannot deliver the list of activities for the following Sol at least the rover will not have completely wasted a Sol. Most of the times this happens whenever the DSN station has a failure (some of their equipment is quite old) or bad weather hits. On one occasion we failed to send commands because at the time of transmission there was an obstacle between Earth and Mars: the Moon!

Do you use some high level frameworks? I mean, do you work with refs, memory directly or write code in declarative way, on the level of operations which should be done, like: «move forward until..», «stop if..», «take stone if it's green..», etc.?

Excellent questions. We have a multi-threaded language but we do not have all the infrastructure typical of multi-threading, for example synchronizing tasks can be challenging. We can query and test some of the internal state of the vehicle (if tilt > 10 deg then stop) but we do not have for, do, or while loops as an example because these could result in an unresponsive system. We can say "move until you reach that position" but at the same time we also say "stop if you moved more than 3 meters". We need to ensure the vehicle is not running freely and responds within a predetermined amount of time.

We can command motion using different levels of complexity. We can go down to the level of "apply 5.7 volts to the left front drive actuator for 4.5 seconds". Or "move the same motor until the wheel has rotated 14 degrees", or "move forward 1.3 meters". We can also go to a much higher level and ask the rover to reach position X, Y while avoiding obstacles and measuring how much the wheel slippage is, oh and while you are doing this, take a DAN measurement every meter or so. We can also tell the rover to reach a particular rock that we see at a specific pixel. We currently do not have the capability to say to the rover "drive to that rock and place MAHLI on it".

In order to code the movement to a tiny stone and picking it up you need great accuracy in coordinates estimation. How is this accuracy achieved?

Our scoop and sieve are not big enough to pick up rocks ;-)

In order to move the vehicle to a very accurate position we use the images we capture with NAVCAMs and determine the distances to various objects, the shape of the terrain, and determine which route is easier to follow for the rover. For example we try to use the smallest number of steps, the lowest number of turns, try to go in straight line segments and on terrain that is as flat as possible. Sometimes we need to be careful not to step over potential science targets, sometimes we have loose rocks to take into consideration. For really precise positioning we also use Visual Odometry. While the vehicle is moving, the software computes the rover position and attitude (roll, pitch and heading) but does not know if the wheels are slipping. VO uses stereo pictures taken by the NAVCAMs to measure the actual motion, including slippage, of the vehicle. This allows the driving sequence to position the rover very accurately, sometimes to within a few

cm! VO is a piece of software that runs on-board and it is a bit slow so we use it only when absolutely necessary.

What kind of motor controllers are used in the robotic arm? Is there a simple PID regulator or you use fuzzy logic or neural networks? Can we reprogram the controllers remotely from the Earth? Do the motor aging and the outer factors are somehow considered? I would like to know more details about the development and testing of low-level hardware for the rover.

I'm not sure if there is a published document about the motor controllers. I will check, one of the rover drivers wrote the code. Motors are brushless, the ones on the arm have encoders and resolvers for redundancy. Motors controllers are implemented on FPGA connected to the CPU. The motion

commands get sent from the CPU to the FPGA which takes care of driving and controlling the actuator. There are a huge number of parameters we can change on the controller such as the PID gains but I'm not sure if we can reprogram the FPGA. Speaking of PID controllers, on Opportunity we tried to clean the mirror of the MiniTES which was lost as a result of the dust storm in 2007. We de-tuned the PID controller of the mirror actuator therefore inducing high frequency oscillations, but that attempt was unfortunately unsuccessful in returning the MiniTES to working conditions.

Do you recall any interesting system bugs you could tell about (considering the NDA)? How long did it take you to fix them?

Except for the Spirit Sol 18 anomaly (see

<http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CDYQFjAA&url=http%3A%2F%2Fwww.cs.princeton.edu%2Fcourses%2Farchive%2Ffall12%2Fcos109%2Fmars.rover.pdf&ei=S3YAUcKyEuSXiAKLsoCYBw&usg=AFQjCNHKNmQp9xXy3cWJj-qJ1IXKpv0Q&bvm=bv.41248874.d.cGE>)

all of the software anomalies were well understood quickly. Fixing a software bug can take a lot of time, not because the fix is difficult but because we need to ensure that the fix does not break anything else. It might take a day or two to implement the fix but many many days with the testbed.

Where can one get the Rover Driver's license? Is perfect understanding of how the rover works required for it? How much training did it take you to finally tell yourself «Well, now it's definitely going to respond perfectly»? Do you feel nervous before sending a new piece of code to the rover? What mistakes should be avoided above all others?

We do not get a real driver's license, nor we need to undergo a formal written or driving test. We sort of train each other and check each other work. If any of us is not confident that something is correct we either read the flight software source code, or test it in the testbed or not do the activity at all. We sometimes refuse to do some activities we believe are too dangerous. Sometimes drivers feel confident but others in the team do not, so we have to come to an agreement on what we think it is safe. Even after 8+ years, 15Km driven on Mars I am never 100% confident all is going to go right. You have no idea how many sleepless night I had in the past few years. ;-)

What do you like most in your job?

Every day there is something new to do, or a new and unusual problem to solve. While some parts of my job are tedious, there is always something new, it is never boring and you learn something new every day.

Do you watch Space Brothers?

No.

How accurately is NASA depicted in the series?

As I never watched any of the episodes, I can't tell. Sorry!

Does Curiosity have some kind of a weapon, like a gun, except for the laser? (for fighting against aliens or other robots)

Obviously not. In fact if we find other life forms or robots we would carefully analyze them, not destroy them!

Would you like to visit Mars personally?

YES!!!! There are some Martian Vistas that I would give anything to see in person. I dreamed about being on Mars several times.

Do aborigines help in avoiding difficult situations, poor roads and quicksand?

Unfortunately the rover moves so slowly that typical solutions like rolling back and forth to get unstuck in mud really cannot be implemented. Also typically on Earth water content in soil is quite different. So the typical driving issues we have on Earth do not apply on Mars. We have instead a group of geologists that are experts on soil mechanics that help us solving some of the more complicated issues such as the Spirit embedding event at Troy.

How did you get a job with NASA? Did it all start with black SUVs blocking every single approach to your house, a couple of Black Hawk helicopters flying above and strange men in black suits knocking at your door saying «you're coming with us!»?

;-) ha ha ha. No black SUVs surrounding me. Many, many years ago I saw an announcement of a position available at JPL. I sent my resume, got invited in for an interview and they liked me and hired me. Several years after I was already working for JPL I was hired to work on MER and then finally on MSL.

How do people generally get employed by NASA?

There is a link on the JPL home page that lists job openings and you can submit your resume. You can also look at the various departments and see if there is a field you are particularly interested and contact the section supervisor directly.

Which of Three Laws of Robotics have been implemented into Curiosity software? What is the amount of code responsible for that Laws?

Since there are no humans on Mars the first law did not need to be implemented. The second and third laws are in a sense really implemented in the software but are dispersed everywhere in many many lines of code. This especially true for the third law of self preservation.

Were there many competitors in the viewing for the job of roverdriver?

Yes, but managed to get into the small group of drivers. The drawback is that getting a similar job elsewhere is impossible! ;-)

If you could, which rover among those you controlled you'd buy? And why?

I wish I could buy Opportunity. She was the first rover I drove on Mars. I still remember my first drive!

Is this a job of your dreams?

Yes! I love to drive and did many road trips in Europe and the USA. Whenever I see a road I always wonder where it leads to. I spend hours on google maps visiting places I will never be able to see. I also love the Martian landscape and even if I see it through the cameras of these rovers it is almost like being there.