A Cloud-based Architecture for Processing 3D Mars Terrain

Parker Abercrombie
Jet Propulsion Laboratory, California Institute of Technology

Curiosity Mars Rover
Mars Reconnaissance Orbiter
On sol 753 of its mission, Curiosity arrived at the base of Mt. Sharp. The foothills at the base of the mountain here are named the Pahrump Hills. These are part of the Murray formation, which underlies all of the other exposed rock layers of Mt. Sharp. Researchers expect that this represents the oldest geological unit that Curiosity will encounter in its journey. Within the Pahrump Hills, Curiosity drilled at three sites, Confidence Hills (sol 759), Mojave (sol 882), and Telegraph Peak (sol 908). At the Mojave site, a preliminary analysis indicated the presence of jarosite, which would point to rock forming in more acidic water that was found in previous sites, and further illustrate the variety in the history of water on Mars. While at the Pahrump Hills, Curiosity took some time to point its cameras up from the rocks below it to observe the comet Siding Spring as it passed closer to Mars.
On sol 753 of its mission, Curiosity arrived at the base of Mt. Sharp. The foothills at the base of the mountain here are named the Pahrump Hills. These are part of the Murray formation, which underlies all of the other exposed rock layers of Mt. Sharp. Researchers expect that this represents the oldest geological unit that Curiosity will encounter in its journey. Within the Pahrump Hills, Curiosity drilled at three sites, Confidence Hills (sol 759), Mojave (sol 882), and Telegraph Peak (sol 908). At the Mojave site, a preliminary analysis indicated the presence of jarosite, which would point to rock forming in more acidic water that was found in previous sites, and further illustrate the variety in the history of water on Mars. While at the Pahrump Hills, Curiosity took some time to point its cameras up from the rocks below it to observe the comet Siding Spring as it passed closer to Mars.
Stats

• ~430 scenes built (and more every day)
• 1 scene = ~1000 input images (5 GB)
• 1 build = 100-300 MB of mesh and texture files
• ~2.5 hr runtime (on 1 computer)
High-level Needs

• Detect when new data is available
• Long running, resource intensive task
• See what’s happening
• Make results available to users
• Bursty workload
Technology Stack*

*Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.
Mars Mission Data System

Build Manager

Jenkins

Worker

OnSight S3 Bucket
1. Poll for new data

Mars Mission Data System

Mars data system

OnSight S3 Bucket
2. Index new data

Mars Mission Data System

Build Manager

Jenkins

Worker

Mars data system

OnSight S3 Bucket
3. Request build

1. Build Manager
2. OnSight S3 Bucket

Mars Mission Data System

Mars data system
4. Manage workers and start job
OnSight S3 Bucket

Mars
Mission Data System

Build Manager

Jenkins

Worker

1. 
2. 
3. 
4. 

Mars data system

5. Pull data and run job

OnSight S3 Bucket
Mars Mission Data System

1. Build Manager
2. S3 Bucket
3. Jenkins
4. Worker
5. Mars data system
6. Upload results

OnSight S3 Bucket
OnSight

S3 Bucket

1. Build Manager
   3. Jenkins
      4. Worker

2. Mars data system

5. Mars Mission Data System

6. Update status

7. OnSight S3 Bucket
1. Build Manager
2. S3 Bucket
3. Jenkins
4. Worker
5. Mars data system
6. Mars data system
7. OnSight
8. Fetch data
Mars data system

Mars Mission Data System

Ubuntu

Build Manager

Jenkins

Worker

Windows

OnSight S3 Bucket
Build Manager
Build Manager

• Purpose:
  • Orchestrates image pipeline builds
  • Dashboard

• Technologies
  • Loopback framework - RESTful API
  • Twitter Bootstrap & Angularjs – frontend
  • RDS - database
### Completed Builds

<table>
<thead>
<tr>
<th>Version</th>
<th>Pipeline</th>
<th>Bucket</th>
<th>Sceneld</th>
<th>SiteName</th>
<th>Status</th>
<th>Enabled</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>201512300855</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>442</td>
<td>sd0052000004</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300848</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>443</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300759</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>442</td>
<td>sd0052000004</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300607</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300525</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300501</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300323</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300306</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512300244</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>20151230023</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>false</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>20151230016</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>20151230016</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>false</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512291307</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>442</td>
<td>sd0052000004</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512291253</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512291248</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512290628</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512290613</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
<tr>
<td>201512290611</td>
<td>0.1.8</td>
<td>landlords-prod</td>
<td>434</td>
<td>sd0052000000</td>
<td>success</td>
<td>true</td>
<td><img src="View" alt="View" /> <img src="Details" alt="Details" /> <img src="Delete" alt="Delete" /></td>
</tr>
</tbody>
</table>
Amazon Relational Database Service (RDS)

- Auto snapshot
- Click to restore
- Zero maintenance required
- Why not NoSQL?
  - Scalability not an issue
  - More concern with capability with existing tools
Data API

- Access data through LoopBack REST API
- Database independent
- Easy to add business logic with JavaScript
CloudWatch Log Management

Event Data
- 2015-12-23T01:00:05.082Z - info: About to scan NCAM for sol 01193
- 2015-12-23T01:00:05.083Z - info: About to scan NCAM for sol 01192
- 2015-12-23T01:00:05.085Z - info: About to scan NCAM for sol 01191
- 2015-12-23T01:00:23.231Z - info: Done processing NCAM
- 2015-12-23T01:00:23.232Z - info: About to scan MCAM for sol 01169
- 2015-12-23T01:00:23.233Z - info: About to scan MCAM for sol 01168
- 2015-12-23T01:00:23.235Z - info: About to scan MCAM for sol 01167
- 2015-12-23T01:00:23.236Z - info: About to scan MCAM for sol 01166
- 2015-12-23T01:00:23.238Z - info: About to scan MCAM for sol 01165
- 2015-12-23T01:00:23.239Z - info: About to scan MCAM for sol 01164
- 2015-12-23T01:00:23.240Z - info: About to scan MCAM for sol 01163
- 2015-12-23T01:00:23.241Z - info: About to scan MCAM for sol 01162
- 2015-12-23T01:00:23.242Z - info: About to scan MCAM for sol 01161
- 2015-12-23T01:00:23.244Z - info: About to scan MCAM for sol 01160
- 2015-12-23T01:00:26.993Z - info: Done processing NCAM
- 2015-12-23T01:00:26.993Z - info: About to scan MCAM for sol 01200
Build Cluster
Jenkins

What is it?
Build system
Continuous Integration & Delivery
**Similar tools**: Bamboo, CruiseControl

What do we use it for?
Manage worker nodes
Compile and run pipeline code

AWS

Amazon SQS  CloudWatch  Amazon EC2  Amazon S3  Amazon EBS
### Build Executor Status

#### master

<table>
<thead>
<tr>
<th>Node</th>
<th>Status</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.1</td>
<td>offline)</td>
<td>#104930</td>
</tr>
<tr>
<td>10.0.0.2</td>
<td>offline)</td>
<td></td>
</tr>
<tr>
<td>10.0.0.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- 10.0.0.1 is offline.
- 10.0.0.2 is offline.
- 10.0.0.3 is occupied by Build_Scene_Prod #1498.

#### Build Queue (1)

- Build_Scene_Staging

### Terrain Build Jobs

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Name</th>
<th>Last Success</th>
<th>Last Failure</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Build_Scene_Staging</td>
<td>1 day 13 hr - #1235</td>
<td>10 days - #1193</td>
<td>2 hr 56 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build_Scene_Prod</td>
<td>1 day 13 hr - #1497</td>
<td>N/A</td>
<td>2 hr 47 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build_Preview</td>
<td>3 mo 20 days - #31</td>
<td>N/A</td>
<td>7 min 50 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build_Generic</td>
<td>12 days - #540</td>
<td>13 days - #539</td>
<td>2 hr 13 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build_CI_Staging</td>
<td>3 mo 1 day - #90</td>
<td>2 mo 27 days - #96</td>
<td>2 hr 35 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Build_CI_Master</td>
<td>12 days - #118</td>
<td>N/A</td>
<td>2 hr 44 min</td>
</tr>
</tbody>
</table>
## Terrain Build Jobs

<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Name</th>
<th>Last Success</th>
<th>Last Failure</th>
<th>Last Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_Scene_Staging</td>
<td>1 day 13 hr - #1235</td>
<td>10 days - #1193</td>
<td>2 hr 58 min</td>
</tr>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_Scene_Prod</td>
<td>1 day 13 hr - #1497</td>
<td>N/A</td>
<td>2 hr 47 min</td>
</tr>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_Preview</td>
<td>3 mo 20 days - #31</td>
<td>N/A</td>
<td>7 min 50 sec</td>
</tr>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_Generic</td>
<td>12 days - #540</td>
<td>13 days - #539</td>
<td>2 hr 13 min</td>
</tr>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_CI_Staging</td>
<td>3 mo 1 day - #90</td>
<td>2 mo 27 days - #96</td>
<td>2 hr 35 min</td>
</tr>
<tr>
<td>☀</td>
<td>☀</td>
<td>Build_CI_Master</td>
<td>12 days - #118</td>
<td>N/A</td>
<td>2 hr 44 min</td>
</tr>
</tbody>
</table>

**Legend**
- RSS for all
- RSS for failures
- RSS for just latest builds

**Build Executor Status**

- master
  - Idle
  - ManageNodes #104930
    - 10.0.0.1 offline
    - 10.0.0.2 offline
    - 10.0.0.3
      - 1 Occupied by Build_Scene_Prod #1498
      - 2 Occupied by Build_Scene_Prod #1498
      - 3 Build_Scene_Prod #1498
      - 4 Occupied by Build_Scene_Prod #1498
Build #1498
(Dec 31, 2015 6:45:30 PM)

Subproject Builds

- Prepare_TerrainSourceAssets #2607
- TerrainToolsLib #4902
- Clean_Result #4293

No changes.

Started by remote host 172.31.27.110

Revision: 77c8cc9a578d70eaa41233a7de70082bef81007f
- origin/production
Build

#1498
(Dec 31, 2015 6:45:30 PM)

Subproject Builds

- Prepare_TerrainSourceAssets #2607
- TerrainToolsLib #4902
- Clean_Result #4293

No changes.

Started by remote host 172.31.27.110

Revision: 77c8cc9a578d70eaa41233a7de70082bef81007f
- origin/production

Started 49 min ago
Build has been executing for 49 min or 10.0.0.1
Progress:
Build #1498
(Dec 31, 2015 6:45:30 PM)

Subproject Builds

- Prepare_TerrainSourceAssets #2607
- TerrainToolsLib #4902
- Clean_Result #4293

No changes.

Started by remote host 172.31.27.110

Revision: 77c8cc9a578d70eaa41233a7de70082bef81007f
- origin/production
Cluster management with Jenkins

Periodic task checks work queue vs. cluster

Work queue

Build Cluster
Cluster management with Jenkins

Work queue

Build Cluster
Cluster management with Jenkins

Work queue

Build Cluster

Task
Handling sub-tasks

Data intensive task, only want one per worker
...But want to be able to run sub-tasks

Worker Node

1
2
3
4
5
6
7

Main task == 4 slots
3 slots left for sub-tasks
Jenkins cluster

- Periodic task to scale cluster
- Automation with Groovy scripts
- Use tags for different type of worker
  - e.g. dev vs. production
Worker Nodes
Workers

- GPU-enabled EC2 machines
- Windows 2012
- Run image processing code (C# binary)
- Created from base image with software installed
# EC2 Instance Types

<table>
<thead>
<tr>
<th></th>
<th>GPU</th>
<th>vCPU</th>
<th>Mem (GB)</th>
<th>Cost*</th>
</tr>
</thead>
<tbody>
<tr>
<td>g2.2xlarge</td>
<td>1</td>
<td>8</td>
<td>15</td>
<td>$0.65/hr</td>
</tr>
<tr>
<td>g2.8xlarge</td>
<td>4</td>
<td>32</td>
<td>60</td>
<td>$2.60/hr</td>
</tr>
</tbody>
</table>

*The cost information contained in this document is of a budgetary and planning nature and is intended for informational purposes only. It does not constitute a commitment on the part of JPL and/or Caltech.*
Worker Lifecycle

Worker Node

1) Pull code from git
2) Pull source data
3) Run pipeline
4) Push to OnSight S3
5) Notify Build Manager

Mission Data System

OnSight S3 Bucket
Workers – Lessons Learned

• Prefer keeping data and code outside of snapshot
• GPU on cloud machines is troublesome
Bonus – Spot Instances

• 7x cheaper!
• $0.10/hr*
• Used for non-critical work

*The cost information contained in this document is of a budgetary and planning nature and is intended for informational purposes only. It does not constitute a commitment on the part of JPL and/or Caltech.
Storage & Distribution
Data Storage & Distribution

• S3 – reliable storage
• CloudFront – distribution network
• Secured via signed cookies
Deployment
Ansible Overview

• IT Automation Tool
  • Software Deployment & Configuration
  • Infrastructure Management (including cloud)
• Similar tools: Chef, Puppet, Salt
Ansible Playbooks

- name: Create jenkins user
  sudo: yes
  user: name=jenkins
Ansible Playbooks

- name: ensure directories permissions
  sudo: yes
  file: path ~/.ssh
  owner=jenkins mode=0700 state=directory
Ansible Playbooks

- name: copy template
sudo: yes
template:
src=jenkins.conf.j2
dest=/etc/default/jenkins
backup=yes mode=0644

notify:
  - restart jenkins
Deploying with Ansible

1) Create EC2 instance

2) `ansible-playbook -i production deploy.yml \   --extra-vars "env=production" \   --vault-password-file VAULT_FILE`
Ansible

• Used to configure all Linux servers
• Not yet using for Windows
Future work

• (More) auto-scaling & management
• Use Ansible to manage worker AMIs
• Split pipeline into smaller services
Acknowledgements

Jeff Norris      Tom Crockett
Jay Torres       Charley Goddard
Alex Menzies     Alice Winter
Jesse Kriss      Matt Clausen
Darren Dao       Microsoft
Q & A

A Cloud-based Architecture for Processing 3D Mars Terrain

Parker Abercrombie
parker.abercrombie@jpl.nasa.gov
Jet Propulsion Laboratory, California Institute of Technology

More info: opslab.jpl.nasa.gov

Feedback: goo.gl/forms/yxhcMrksaS