HPC Track
CEI: Beyond EnSight

Darin McKinnis, VP Sales and Marketing
EnSight

Leading HPC post-processor

High-quality rendering
High Performance
Interactivity and Batch Operations
Extensive feature set
Visualization and Calculation Capabilities
  Variable Calculator
  Plotting
General Purpose
  Multi-physics
  Mix solver results
  FEA, CFD, EMAG, Particles, etc.
Python Extensibility
Multi-case Support
  32 cases at one time
  4 cases cloned
HPC Capacity – Client-server and Distributed Arch
Virtual Reality - Caves, Walls, Headsets
EnSight 10.2

Release: October 2016

Focus on

Enhanced graphics performance
- Vertex Buffer Objects (VBOs) and LIC

Enhanced graphics realism
- Materials Library
- Additional Lighting Models
- Integrated Ray-tracing output

Co-processing

Supports Knowledge Capture Triggers and Events
Overcome Problems with Current Post-processing

Output is Images, Movies, Plots, Values without Connections

Loss of Provenance

Single system for some customers

Provide a connection to other Tools/Systems

Caching/Compatibility system for other vendor’s systems
Provide high-quality, interactive output from information gleaned from multiple EnSight sessions and other sources (e.g. solver)

- Support for multiple deployment scenarios
  - Isolated user (local), Departmental, Cloud (both as a source and target)
- Infrastructure for handling data from multiple sessions
  - Parameter studies, Longitudinal studies, QA/QC, etc.
    - Ability to coalesce data from different datasets/runs into composite visuals
    - Playing a larger role in data stewardship
- Leverage framework advances ushered in by the move to the Cloud
  - Web-orientated toolsets
    - High-quality plotting packages, portable animation formats, 3D Geometry
    - JavaScript, HTML5, WebGL, etc
- Customizable to integrate with existing workflows
  - Value-add (above EnSight) data products that can be externally integrated
Key Features

• Capture: mechanism to collect and organize “data items”
  • Standard elements: Text, HTML, 3D Geometry, Tables, Images, Movies
    – User/Tool/Site customizable
  • Automated and user activated capture “triggers”
• Common item storage
  • Database storage, local or remote
  • Provenance specified via context
    – User tags, application information, dataset
    – Watermarking
• Report generation
  • Hierarchical templates
    – Item selection/filtering mechanism
    – Process (think pivot tables) and display elements
  • HTML output, potentially other targets
**Terminology**

Data Item (Items)
- A basic unit of information that is stored in the database
- Focus on raw data, report framework provides visual representation

Source
- EnSight Python extension that produces report items
- Each instance can have independent parameters

Trigger
- Cause one or more sources to generate items, placing them in the database
- EnSight Python extension: Python/command language, interactive & automatic triggers
### Run-time Data Ingest

#### Global Product Data Interoperability Summit | 2016

<table>
<thead>
<tr>
<th>Report Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session</td>
<td>550104fe-26a7-11e6-99...</td>
</tr>
<tr>
<td>Dataset</td>
<td>8245ee01-58d5-3347-91...</td>
</tr>
<tr>
<td>TimeDate</td>
<td>2016-05-30 16:53:39.696566</td>
</tr>
<tr>
<td>UserTags</td>
<td>DataLoad, Run Number</td>
</tr>
<tr>
<td>Source</td>
<td>Image Source</td>
</tr>
<tr>
<td>Type</td>
<td>Image</td>
</tr>
</tbody>
</table>

#### System Components

- **Item Database**
- **REST Interface**
- **Web Server**
- **Report Generation Framework**

### EnSight Interface

- **Data Load**
- **Image Source**
- **Custom HTML**

### Dataset Details

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reader</td>
<td>Case</td>
</tr>
<tr>
<td>UserTags</td>
<td>My dataset,...</td>
</tr>
<tr>
<td>Filename</td>
<td>foo.case</td>
</tr>
<tr>
<td>Applicaton</td>
<td>EnSight</td>
</tr>
<tr>
<td>Version</td>
<td>10.2.0(a)</td>
</tr>
<tr>
<td>Platform</td>
<td>win64</td>
</tr>
</tbody>
</table>

### Additional Information

- **Session**: 550104fe-26a7-11e6-99...
- **Applicaton**: EnSight
- **Version**: 10.2.0(a)
- **Platform**: win64
Run-time Data Ingest

Global Product Data Interoperability Summit | 2016

Solver
Fluent/ANSYS/LS-Dyna/etc

Solution Information Export Script (Python)

EnSight

Data Load

Image Source

Custom HTML

Report Item

<table>
<thead>
<tr>
<th>Report Item</th>
<th>Dataset</th>
<th>Session</th>
<th>TimeDate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>13ce7f9a-abb5-4384-80…</td>
<td>9b11802c-e640-4b08-bf…</td>
<td>2016-05-28 13:33:23.45434</td>
</tr>
</tbody>
</table>

UserTags
Fluent, Inputdeck, Run Number 10, ...

Source
Fluent Input Deck Source

Type
Text

Dataset

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Reader</th>
<th>UserTags</th>
<th>Application</th>
<th>Version</th>
<th>Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>13ce7f9a-abb5-4384-80…</td>
<td>Fluent</td>
<td>Run Number 10</td>
<td>Fluent</td>
<td>17.0</td>
<td>win64</td>
</tr>
</tbody>
</table>

Runnable Time Data Ingest

Data Load

Image Source

Custom HTML

User Trigger

Table

Source

Image

Report Generation Framework

Web Server

REST Interface

Item Database

Solution Information Export Script (Python)
Start with a general search, select all items included in the report

Reports are a hierarchy of templates
  - Each templates includes a filtering/sorting operation
    - Select what items are to be passed to which child templates
  - Child templates can be repeated
    - Example: parent template categorizes all input items by crank angle. The child template is then invoked repeatedly, each time with the subset of items for that specific angle.
  - Generator templates compute new items from input items
    - New items are not stored in the database
    - Force specific visual appearance
      - Examples: force as pie chart, change background color, label formatting, etc
    - Generate new data
      - Examples: collect all rows with a specific name from all input table items and make a new table. Collapse all columns in the input table into a table of maximum values.

Leaf nodes generate HTML output as per current rendering schemes
Deployment: Local Desktop
Deployment: Departmental Server

Interactive User
- EnSight Interactive client

Interactive User
- Web Browser

Departmental HPC Server
- Analytics
- EnSight Batch mode

Departmental Web Server
- Report Generation Framework
- Web Framework
- Local files

Departmental Database Server
- PostgresQL Database
Deployment: Cloud (AWS) Server

Interactive User
- EnSight Interactive client

Interactive User
- Web Browser

Departmental HPC Server
- Analytics
- EnSight Batch mode

AWS Servers
- Web Server Instance
  - Report Generation Framework
- Database Server Instance
  - PostgresQL Database
- Filestore Server Instance
  - EBS/S3 "files"

EnSight
- Interactive client
- Batch mode

Analytics
- Interactive user
- Web Browser

Web Browser
- Interactive User
- Web Server Instance
- Web Framework

EnSight Batch mode
- Compute (EC2) Instance
- Analytics
The Tech Stack

Global Product Data Interoperability Summit | 2016

• Target platform
  • HTML5 (canvas) and WebGL
  • Chrome, Firefox, IE11, Edge, Safari, Qt5 WebKit
• OpenSource components
  • 3D Geometry
    – Babylon.js/WebGL, simple export demonstrated
  • Plotting
    – plotly.js (recently OpenSource)
• Web Framework
  – Django + Bootstrap.js + Apache
• Databases
  – PostgreSQL, SQLite
• Cloud/distributed deployment
  • AWS, custom (departmental, local)
The Steps Ahead…

Currently in a prototyping stage
  - Core EnSight extensions in place
    - All Python extensions are potential data sources and triggers
    - Report item and Trigger interfaces prototyped
  - New technologies
    - Improved mp4 (H.264) support (display in all major browsers)
    - 3D Geometry prototype integrated (babylon.js)
    - Basic plotting integration complete: line, bar, pie (plotly.js)
    - Django web framework, SQLite and PostgreSQL backend databases tested
    - Watermarked imagery

Installer in place
  - Core extensions built into EnSight
  - Installer for server framework
    - Part of EnSight 10.2
## Notional Timeline

**Data Capture**
- Core complete
- Toolset selected
- Prototype running

**Triggers and Sources**
- Basics complete
- Discussion with CEI “insiders”
- Define shipping modules
- Command language/Python API definition

**Report**
- Early design phase
- Standalone report template editing tool
- Filtering system
- Define shipping modules
- Customer exposure/discussion?

**Tuning**
- Not started
- Performance
- Security
- 3D Geometry revisited
- Customer testing?

**Release**
- Not started
- Documentation
- Licensing

---

*Copyright © 2016 CEI - Confidential & Proprietary Information*
• Thank you
Web Nouns and Verbs

Global Product Data Interoperability Summit | 2016

- JavaScript - ECMAScript 6
  - The programming language of the browser
  - JSON – JavaScript Object Notation
- REST - Representational state transfer
  - APIs over HTTP(S)
- Web Server – Apache, Nginx
  - Handles incoming REST formatted requests vs URIs
    - scheme:[//[user:password@]host[:port]][/]path[?query][#fragment]
  - Can dispatch to other frameworks
- Web Framework
  - Maps REST (URLs + action + payloads) to code that generates web pages
  - Django
- ORM - Object-relational mapping
  - Makes a database (e.g. SQL) look like objects
The Pipes and Tubes

SQL Database
PostGIS, SQLite

WebServer
Apache, Nginx
Handles HTTP, proxy redirects, Authentication

WebFramework
Django
Data model objects
Generates HTML/JSON/etc

ORM

WebBrowser
Chrome, IE, Safari

HTML5
WebGL
JavaScript

Filestore
S3, local disk

HTML/JSON/files

URI + JSON

Copyright © 2016 Boeing. All rights reserved.

Copyright © 2016 Northrop Grumman Corporation. All rights reserved.

GPDIS_2016.ppt | 22