Web Services
and their benefits to Central Cancer Registries

Fabian Depry, IMS

NAACCR 2013, Austin, Texas
In Today's Presentation

• Definitions

• Real-life Examples
  o Example 1: Online Time Services
  o Example 2: SEER API

• Web Services and Central Registries
  o Pros and Cons of Web Services
  o Security

• Use Cases
  o Use Case 1: Abstracting Tools
  o Use Case 2: Hospital Tools

• SEER*DMS Web Services
Definitions

• From Wikipedia:
  A web service is a method of communication between two electronic devices over the World Wide Web.

• Less formally:
  A web address (URL) that can be called from software to get shared data or computation results.
Example 1: Online Time Services

- Returns the current time
- Publicly available (no security required).
- Can be accessed using a web browser, but it's not the main purpose.
- The call returns a value that can be used in other software.
Example 2: SEER API

SEER API is a secure Web Service:

After providing my key:

```json
{"code":401,"message":"You must supply an API key"}
```

```json
[{
  "number":1,"name":"LipUpper"},
  {"number":2,"name":"MelanomaLipUpper"},
  {"number":3,"name":"MelanomaLipLower"},
  {"number":4,"name":"MelanomaLipOther"},
  {"number":5,"name":"LipOther"},
  {"number":6,"name":"MelanomaLipOther"},
  {"number":7,"name":"TongueBase"},
  {"number":8,"name":"MelanomaTongueBase"},
  {"number":9,"name":"TongueAnterior"},
  {"number":10,"name":"MelanomaTongueAnterior"},
  {"number":11,"name":"GumUpper"},
  {"number":12,"name":"MelanomaGumUpper"},
  {"number":13,"name":"GumLower"},
  {"number":14,"name":"MelanomaGumLower"},
  {"number":15,"name":"GumOther"},
  {"number":16,"name":"MelanomaGumOther"},
  {"number":17,"name":"FloorMouth"},
]```
Example 2: SEER API (cont)

The SEER*CS Test Web Site is an example of software using the SEER API services:

[Image of the SEER*CS Test Web Site]
Pros and Cons of Web Services

• Pros:
  o Fully integrated component of any software
  o Increased security - controlled by the system instead of the users
  o Reduces file creation and storage
  o Allows better data tracking
  o Can start with simple services and build more complex ones over time

• Cons:
  o Requires advanced IT skills to put in place
Security

• Essential for any cancer registry.
• Secured Web Services can be implemented by the service providers --> improved security because user's responsibility is reduced.
• The technology offers many ways of adding secured services.
• Web Services allow call tracking, which is a big aspect of secured services.
Use Case 1: Abstracting Tools

- Lookups
- Physicians
- Facilities
- Casefinding List
- Reference Data

Abstracting Tool (Field Laptop)

Cancer Registry

- Abstracted Cases
- Non Abstracted Cases
Abstracting Tools (cont)
Use Case 2: Hospital Tools

Cancer Registry

Get Patient Information Provided by this Facility

Update Patient Information

Hospital, Lab, etc...
Hospital Tools (cont)

![Image of Louisiana Tumor Registry Data Management System](image)

<table>
<thead>
<tr>
<th>Display Id</th>
<th>Last Name</th>
<th>First Name</th>
<th>Social Security Number</th>
<th>Birthdate</th>
<th>Sex</th>
<th>Race</th>
<th>Last Contact</th>
<th>Vital Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAT-00353037</td>
<td>DUCK</td>
<td>DAFFY</td>
<td>9999999874</td>
<td>99/99/1902</td>
<td>01</td>
<td>8/21/1993</td>
<td>Dead</td>
<td></td>
</tr>
<tr>
<td>PAT-20060687</td>
<td>DUCK</td>
<td>DONALD</td>
<td>123456789</td>
<td>99/99/1985</td>
<td>01</td>
<td>3/1/2012</td>
<td>Alive</td>
<td></td>
</tr>
</tbody>
</table>
SEER*DMS Web Services

Introduction
Privacy unencrypted HTTP is not supported.

Contact
GET api/contact/displayId/:displayId
GET api/contact/filters
GET api/contact/id/:id
GET api/contact/search

Data Search
GET api/dataset
GET api/dataset/id/:id

Facility
GET api/facility/displayId/:displayId
GET api/facility/filters
GET api/facility/id/:id
GET api/facility/search

Lookup
GET api/lookup
GET api/lookup/id/:id

Patient
GET api/patient/displayId/:displayId
GET api/patient/filters
GET api/patient/id/:id
GET api/patient/search

Record

Example
One of the strengths of REST-style web services is that they are both platform and language agnostic. As long as a programming language can make standard HTTP calls and parse the results then they have all they need to access the web service.
## SEER*DMS Web Services (cont)

### GET api/facility/search

Return a list of Facilities based on various criteria.

**URL Format**
https://seerdms-h1-dev.lmsweb.com/seerdms_h1/api/facility/search

**Permission**
api_facility_read

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Required</th>
<th>Type</th>
<th>Default Value</th>
<th>Example Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Facility Filter ID. If no id is provided, it will return all the facilities.</td>
<td>no</td>
<td>String</td>
<td></td>
<td></td>
</tr>
<tr>
<td>activeOnly</td>
<td>Pass 'true' to only return active facilities, and 'false' for inactive facilities.</td>
<td>no</td>
<td>Boolean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>depth</td>
<td>Level of detail to return:</td>
<td>no</td>
<td>String</td>
<td>IDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Only includes facility id, display id and name</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- All Facility information</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ignore</td>
<td>List of properties which will not be returned with the Facility. This parameter can be supplied multiple times to specify a list of properties. The help interface will automatically split on commas and supply multiple parameters.</td>
<td>no</td>
<td>[String]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>start</td>
<td>Index of the first result.</td>
<td>no</td>
<td>Integer</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>count</td>
<td>Maximum number of Facilities to return per API call. If specified, value must be between 1 to 100.</td>
<td>no</td>
<td>Integer</td>
<td>100</td>
<td>10</td>
</tr>
</tbody>
</table>

[Run]
Conclusion

For transferring data, Web Services beat most of the traditional methods:
• They can be seamlessly added to an existing Central Registry's software
• They offer more security and tracking options

If your registry doesn't currently provide Web Services, ask your IT staff to look into it 😊