An Empirical Study on the use of Net-Centric Services (NCS) in USC e-service projects

Kamonphop Srisopha
Annual Research Review (ARR)
April 4, 2017
USC e-service projects

- Various types of project proposal from within the USC community, local LA small businesses, non-profit organization, and government agencies.
- 6 - 18 projects are selected each year.

For example:
- “VYMA” (Verdugo Young Musician Organization) website renovation project
- “Farm Worker” project proposal from a former California State Senator Dean Florez
- “LINGGGO” project proposal from Dr. Zsuzsa from USC Dornsife.
- “Personnel Department-City of Los Angeles mobile application” - job seeker on mobile for City of LA Personnel Department
Non-Development Items (NDI)
(COTs/GOTs/ROTs/OSS/NCSs)
“A given item that is acquired from a third party or outside organization, which is not designed specifically for your use, and you have no control over how the item will evolve in the future”

- Dr. Chris Abts

System NDI:

Application NDI:

Both blackbox/whitebox
Net-Centric Services

(Specifically, [ ] as a service (aka “on demand”) - black box)

- Firebase
- Google Maps
- Amazon Web Services
- Stripe
- Salesforce
- PayPal
- Nexmo
- Twilio
- Squarespace
- Disqus
- MailChimp
- SurveyMonkey
- Qualtrics

Customer relationship management (CRM)

And many many more ....
Global Cloud Services Trend

- Can we observe a similar trend in the adoption of NCS in USC e-service projects over the years?
- How does using NCSs relate to development effort and project size?

Source: Cisco global cloud index, 2013-2018
Source: Centaur Partners
Research Questions

**RQ1:** Is there an upward trend in the use of NCSs in USC e-service projects?

**RQ2:** Does development effort vary between projects that use NCSs for end-user functionality and projects that do not?

**RQ3:** Does SLOC vary between projects that use NCSs for end-user functionality and projects that do not?
RQ1: Is there an upward trend in the use of NCSs in USC e-services projects?

Steps:

Each project team has to document their NCS/NDI usage/choices in their FED, OCD, and SSAD reports.

1. We count the number of NCSs use from those documents for each team.

1. We combine our data (6 years) with the data excerpted from Dr. Supannika’s dissertation (data from 2005-2009).
An Increasing trend in the use of NCSs over the years
Projects that decide to use NCS will use multiple NCSs
- Trend is logarithmic meaning it is slowly increasing and will eventually stabilize
Top 3

1. Geolocation services
   a. (e.g., Google Maps, Google Place, MS Bing Maps)

2. Financial services
   a. (e.g., Paypal, Stripe)

3. Social Network APIs
   a. (e.g., Facebook (login with facebook, FB platform), Twitter, LinkedIn)
RQ2: Does **development effort** vary between projects that use NCSs for end-user functionality and projects that do not?

**Steps:**

Developers have to log their effort in a project management/issue tracking software (e.g., Jira)

1. We retrieve the effort data* only the ones that relate to **development effort**

1. We sum the effort of each developer who works in the project to get the overall development effort of the project*.

*the data is based on one-semester project team from 2015 and 2016
One-way ANOVA for Effort

<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>F = 6.19545</th>
</tr>
</thead>
<tbody>
<tr>
<td>Between-treatments</td>
<td>118517.6171</td>
<td>1</td>
<td>118517.6171</td>
<td></td>
</tr>
<tr>
<td>Within-treatments</td>
<td>191297.7935</td>
<td>10</td>
<td>19129.7793</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>309815.4106</td>
<td>11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The $f$-ratio value is 6.19545. The $p$-value is .032041. The result is significant at $p < .05$.

$p$-value = 0.032 < 0.05

- There is a significant difference between effort for teams that use NCS for end-user functionality and teams that do not.
Difference in effort between projects that use NCSs for end-user functionality and projects that do not

The result supports intuition that using NCSs = lower development effort
RQ3: Does SLOC vary between projects that use NCSs for end-user functionality and projects that do not?

Steps to retrieve actual SLOC:

1. Run UCC* to get the total SLOC of the project source code.
2. Manually examine the source code and inspect the team’s documentation.
3. Note down the version of the NDI(s) that the team is using.
4. Download those NDI(s) and count* its SLOC.
5. Apply the equation below.

\[
\text{Actual SLOC} = \text{Total SLOC} - \sum \text{NDI SLOC}
\]

*Unified Code Count (UCC) is used to count source lines of code.
One-way ANOVA for SLOC

<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Between-treatments</td>
<td>9146304.4675</td>
<td>1</td>
<td>9146304.4675</td>
<td>13.1988</td>
</tr>
<tr>
<td>Within-treatments</td>
<td>6239957.7143</td>
<td>9</td>
<td>693328.6349</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>15386262.1818</td>
<td>10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The f-ratio value is 13.19188. The p-value is .005468. The result is significant at p < .05.

p-value = 0.005468 < 0.05

- There is a significant difference between SLOC for teams that use NCS for end-user functionality and teams that do not.
Difference in SLOC between projects that use NCSs for end-user functionality and projects that do not.

The result supports intuition that using NCSs = smaller project size.
Estimated* vs Actual size

Relatively higher over estimation of project size

Projects that use NCS

Projects that do not

*Based on the project’s COCOMO II data
Conclusion

• Based on USC e-services projects data from 2005-2016, there is an increasing trend in the use of NCSs over the years.

• We employed one-way ANOVA to investigate:
  • RQ2: Does development effort vary between projects that use NCSs and project that do not?
    • Yes, it is different with statistical significance.
    • Projects that use NCSs have relatively lower development effort.

  • RQ3: Does SLOC vary between projects that use NCSs and project that do not?
    • Yes, it is different with statistical significance.
    • Projects that use NCSs have relatively smaller SLOC
    • Over estimation for project size
Future Work

Some interesting RQs:

• Does using NDIs/COTs (as a black box) produce the same phenomena?

• How does one accurately estimate the effort and size of such project? (calibrate COCOMO II, COCOTS)

• How about estimating the migration cost?

• Does using NCSs make project more successful?

• Case study on projects that depend on NCS that is no longer available (e.g., Parse.com)

• How do we measure maintainability level/technical debt of such project?