# SERVICE INTERACTION NETWORKS MODELING

*Author: Leila Zamani*
*Supervisor: Prof. Joseph Davis*
School of IT
FACULTY OF ENGINEERING & INFORMATION TECHNOLOGIES

## Aims:

In this research project, we propose a model of service interaction networks (SIN) in service delivery and investigate the critical factors that characterise the interactions among the network agents as they impact on the performance of the system.

## Introduction:

Service systems rely on internal interactions among service provider agents and the external interactions with customer agents in the design and delivery of services. The analysis and modelling of these interactions are essential to the design of effective and efficient service systems. In a typical service delivery system, each service request is typically performed by a set of the actors based on the negotiated requirements. This setting of service delivery process is viewed as an abstract network and the collection of agents representing various tasks and their interactions and the outcomes is referred to as service interaction networks [1].

This research focuses on service interaction networks (SIN) in the context of the design and delivery of complex, IT-centric services.

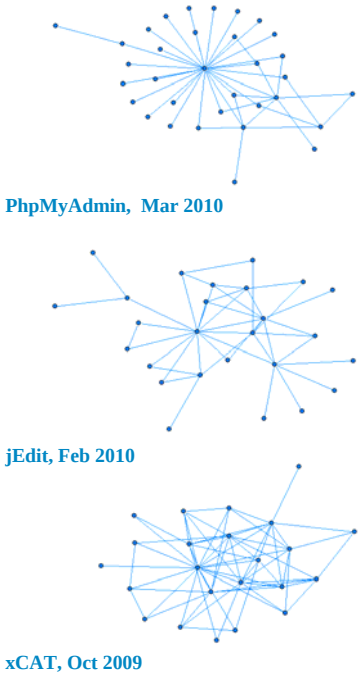


***Figure 2: Interaction Networks Structures for three OSS Projects in a specific month***

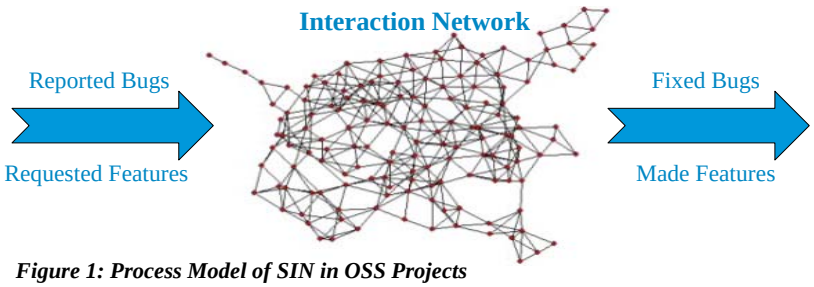


***Figure 1: Process Model of SIN in OSS Projects***

## Research Questions:

The proposed model is based on the idea that certain structural properties of service interaction networks can have significant effect on the effectiveness of the service system (*Figure 1).*

In this model, we explore network degree centrality, eigenvector centrality and network size, as they impact on the effectiveness of service interaction networks. We also examine service delivery time, rate of task completion, average number of error reports or new requests and frequency of usage as the measures of effectiveness in service interaction networks.

## Method:

In order to examine the validity of the proposed model, we test it with real world data. Open source software (OSS) projects provide good proxies for the interactions between service provider and customer agents, similar to developers and users, to fix bugs or develop new features *(Figure 2).*

Statistical analysis based on the acquired data from OSS projects has been used to test the hypotheses.

## Data:

Data on a sample of 50 projects over one year period *(May 2009 to April 2010)* are acquired from SourceForge.net [2].

These projects are chosen from a variety of project categories and based on three factors: project status, project rank, and the number of developers.

## Results:

The results show that the degree centrality has a strong negative effect on the number of error reports or new requests. Moreover, the size of the networks positively influences the rate of task completion. *(Table 1)*

| | NoRequests |
|---|---|
| OutDegree | -0.123** |
| | (-3.08) |
| _cons | 36.11*** |
| | (27.23) |
| t statistics in parentheses | |
| *p<0.05, **p<0.01, ***p<0.001 | |

| | NoRequests |
|---|---|
| InDegree | -0.120** |
| | (-3.06) |
| _cons | 36.25*** |
| | (25.27) |
| t statistics in parentheses | |
| *p<0.05, **p<0.01, ***p<0.001 | |

| | TaskCompletion |
|---|---|
| NoDeveloper | 0.033** |
| | (2.72) |
| _cons | -0.0361 |
| | (-0.11) |
| t statistics in parentheses | |
| *p<0.05, **p<0.01, ***p<0.001 | |

***Table 1: Statistical results based on the data from 50 OSS Projects over the period of May 2009 - April 2010***

## Future Work:

- Extend the dataset to 100 OSS projects and re-examine the proposed hypotheses.
- Investigate the effects of other network issues (e.g. network life cycle) on the effectiveness of service interaction networks.
- Extend the model to capture behavioral parameters (e.g. task ownership or interests, task learning, etc.) related to service providers.

## References:

*[1] S. Kameshwaran, et al., "Analyses for service interaction networks with application to service delivery", in Proceeding of SIAM International Conference on Data Mining, 2009.*

*[2] SourceForge.net, The world 's largest OSS development website.*