Data marketplaces: core models and concepts

Hong-Linh Truong
Distributed Systems Group, TU Wien

truong@dsg.tuwien.ac.at
http://dsg.tuwien.ac.at/staff/truong
@linhsolar
Outline

- Data marketplaces
- Description models
- Data contract model and evaluation
- Data contract exchange models and architectures
- Case study – design a data marketplace
Recall – data service units in clouds/internet
Recall – data as a service

Data-as-a-Service – service models

- Data publish/subcription middleware as a service
- Sensor-as-a-Service

- Database-as-a-Service (Structured/non-structured querying systems)
- Storage-as-a-Service (Basic storage functions)

Edge and/or Cloud Systems
Data platform or marketplace?

http://www.guavus.com/platform/

The Guavus Reflex® Platform

Microsoft Azure Marketplace

http://datamarket.azure.com/browse/data

ASE Summer 2016
Data marketplaces

- More than just DaaS
  - DaaS focuses on data provisioning features
- Stakeholders in data marketplaces
  - Multiple data providers and consumers
  - Marketplace providers
  - Marketplace authorities
  - Analytics providers
  - Data transportation providers
  - Billing and payment providers
Example of stakeholders

Specific data market (Tokyo Tsukiji) or generic data market (Donau Zentrum)?
Technical services, protocols, mechanisms in data marketplaces

- Multiple DaaS provisioning
  - Access models and interfaces
- Complex interactions among DaaS providers, data providers, data consumers, marketplace providers, etc.
  - Data exchange as well as payment
- Complex billing and pricing models
- Market dynamics
- Service and data contracts
Some important issues

DAAS DESCRIPTION MODEL

DATA CONTRACT

DATA CONTRACT EXCHANGE
Description Model for DaaS (1)

Which levels must be covered?

![Diagram showing the levels of a DaaS model with connections between consumer, SOAP/REST APIs, DaaS, and data resources.]
Description model for DaaS – types of information

Which types of information must be covered?

- Quality of data
- Ownership
- Price
- License
- Service interface
- Service license
- Quality of service
- ....
- ....
DEMOS – a description model for Data-as-a-Service

Quang Hieu Vu, Tran Vu Pham, Hong Linh Truong, Schahram Dustdar, Rasool Asal: DEMODS: A Description Model for Data-as-a-Service. AINA 2012: 605-612

See prototype: http://www.infosys.tuwien.ac.at/prototype/SOD1/demods/
Data marketplaces and related components/services
Data contract

How to specify data contract?
Data contracts

- Give a clear information about data usage
- Have a remedy against the consumer for illegal data usage
- Limit the liability of data providers in case of failure of the provided data;
- Specify information on data delivery, acceptance, and payment
Data contracts

- Well-researched contracts for services but not for DaaS and data marketplaces
  - But service APIs != data APIs =! data assets
- Several open questions
  - Right to use data? Quality of data in the data agreement? Search based on data contract? Etc.

→ Require extensible models
  → Capture contractual terms for data contracts
  → Support (semi-)automatic data service/data selection techniques.

Study of main data contract terms

- Data rights
  - Derivation, Collection, Reproduction, Attribution
- Quality of Data (QoD)
  - Not mentioned, Not clear how to establish QoD metrics
- Regulatory Compliance
  - Sarbanes-Oxley, EU data protection directive, etc.
- Pricing model
  - Different models, pricing for data APIs and for data assets
- Control and Relationship
  - Evolution terms, support terms, limitation of liability, etc.

Most information is in human-readable form

ASE Summer 2016
## Data contract study

<table>
<thead>
<tr>
<th>Data contract study</th>
<th>Data contract study</th>
<th>Data contract study</th>
<th>Data contract study</th>
<th>Data contract study</th>
<th>Data contract study</th>
<th>Data contract study</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data rights</td>
<td>Quality of data</td>
<td>Compliance</td>
<td>Pricing model</td>
<td>Control and relationship</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Derivation</td>
<td>Collection</td>
<td>Reproduction</td>
<td>Attribution</td>
<td>Non-commercial use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transaction</td>
<td>Subscription</td>
<td>Warranty</td>
<td>Indemnity</td>
<td>Liability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Laws, Jurisdiction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

Developing data contracts in cloud-based data marketplaces

- Follow community-based approach for data contract
- Propose generic structures to represent data contract terms and abstract data contracts
- Develop frameworks for data contract applications
- Incorporate data contracts into data-as-a-service description
- Develop data contract applications
Community view on data contract development

- Community users can develop:
  - Term categories, term names, values, and units
  - Rules for data contracts
  - Common contract and contract fragments

Community users $\neq$ novice users
Representing data contract terms

- **Contract term**: (termName, termValue)
  - **Term name**: common terms or user-specific terms
  - **Term value**: a single value, a set, or a range

<table>
<thead>
<tr>
<th>Category</th>
<th>Term representation</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data rights</td>
<td><code>termName</code></td>
<td><code>termName</code>={Derivation, Collection, Reproduction, Attribution, Noncommercialuse}, <code>val_i</code> ={Undefined, Null, Allowed, Required, True, False}</td>
</tr>
<tr>
<td></td>
<td><code>= \{val_1, val_2, \ldots, val_n\}</code></td>
<td></td>
</tr>
<tr>
<td>Quality of data</td>
<td><code>val_l \leq termName \leq val_u</code></td>
<td><code>termName</code>={Accuracy, Completeness, Uptodateness}, <code>val_l</code> and <code>val_u</code>\in[0, 1]</td>
</tr>
<tr>
<td>Compliance</td>
<td><code>termName</code></td>
<td><code>termName</code> and <code>val_i</code> are any string, e.g., <code>termName</code>={PrivacyCompliance} and <code>termValue</code>={Sarbanes-Oxley (SOX) Act}</td>
</tr>
<tr>
<td></td>
<td><code>= \{val_1, val_2, \ldots, val_n\}</code></td>
<td></td>
</tr>
<tr>
<td>Pricing model</td>
<td><code>termName</code></td>
<td><code>termName</code> is any string, e.g., MonthlyPayment; <code>val_1</code>\in R, e.g., <code>cost</code> = 50 €, <code>val_2</code> = <code>\{(\text{end}_i - \text{start}_i); UNLIMITED\}</code></td>
</tr>
<tr>
<td></td>
<td><code>=(cost = val_1, usagetime = val_2, maximumuse = val_3)</code></td>
<td><code>where \text{end}_i, \text{start}_i</code>\in datetime, e.g., <code>usagetime</code> = 30 days; <code>val_3</code>\in N, e.g., <code>maximumuse</code> = 1,000 calls</td>
</tr>
<tr>
<td>Control and relationship</td>
<td><code>termName = val</code></td>
<td><code>termName</code> and <code>val</code> are any string, e.g., <code>termName</code>={Liability, LawAndJurisdiction} and <code>val</code> = <code>{US, Austria}</code></td>
</tr>
</tbody>
</table>
Structuring abstract data contracts

Concrete data contracts can be in RDF, XML or JSON

Use Identifiers and Tags for identifying and searches
Development of contract applications

- Main applications:
  - Data contract compatibility evaluation, data contract composition

- Some common steps
  - Extract DCTermType in TermCategoryType
    - Extract comparable terms from all contracts,
      - e.g., dataRight: Derivation, Composition and Reproduction
  - Use evaluation rules associated with DCTermType from rule repositories
  - Execute rules by passing comparable terms to rules
  - Aggregate results
Evaluating Data Contracts

- **Goal**
  - Check the quality and reputation of a data contract

- **We can check data contracts using quality of data metrics**
  - Timeliness, Completeness, Reputation, Consistency metrics

- **Examples**
  - Free-per-use but cost = 100EUR
  - Missing „data accuracy“ concern
Goal
If multiple data contracts are compatible with the consumer needs
→ The consumer requires multiple data associated with different contracts

Contract compatibility
- Matching contract terms
- Evaluating contract term compatibility and completeness w.r.t. application needs
- Making decision in using data
Example of contract compatibility evaluation

<table>
<thead>
<tr>
<th>Action</th>
<th>Case</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compatibility = 1</td>
<td>Check reputation, consistency and timeliness</td>
<td>Reputation = LOW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Consistency = LOW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Timeliness = LOW</td>
</tr>
<tr>
<td>Compatibility &lt; 1</td>
<td>Check completeness and timeliness</td>
<td>Completeness = LOW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Timeliness = LOW</td>
</tr>
</tbody>
</table>
Conceptual architecture for contract management and evaluation

- Prototype
  - RDF for representing term categories, term names, term values, units
  - Allegro Graph for storing contract knowledge
Illustrating examples

- A large sustainability monitoring data platform shows how green buildings are
  - Real-time total and per capita of CO2 emission of monitored building
  - Open government data about CO2 per capita at national level
- We created contracts from
  - Open Data Commons Attribution License
  - Open Government License
Existing common knowledge about Open Data Commons
Step 2: provide OpenBuildingCO2

OpenBuildingCO2 by modifying quality of data and data right

OpenGov for government data

Data contract for green building data
Example – composing data contract terms

```
PREFIX adcm: <http://www.infosys.tuwien.ac.at/SOD1/adcm#> 

CONSTRUCT { 
}

WHERE { 
  ?varAcc1 rdf:type adcm:SingleValueExpressionType .
  ?varAcc1 adcm:numericValue ?value .
  ?varAcc1 adcm:binaryOperator adcm:atLeast .
  ?varAcc2 rdf:type adcm:RangeExpressionType .
  FILTER (?value <= ?upperBound) .
  LET (?compositeLowerBound := afn:max(?value, ?lowerBound)) .
  LET (?compositeUpperBound := ?upperBound) .
}
```

<table>
<thead>
<tr>
<th>Subject</th>
<th>[Predicate]</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>ns1:compositeAccuracy</td>
<td>ns1:lowerBound</td>
<td>0.7</td>
</tr>
<tr>
<td>ns1:compositeAccuracy</td>
<td>ns1:upperBound</td>
<td>0.95</td>
</tr>
</tbody>
</table>
Discussion time

CAN WE AUTOMATICALLY GENERATE DATA CONTRACTS FOR NEAR-REALTIME DATA?
Exchange data contract (1)

How do they interact w.r.t. data concerns?
How do their data contracts look like?
Exchange data contract (2)

- Lack of models and protocols for data contract exchange in data marketplaces
  - Constraints for data usage are not clear
  - Inadequate data/service description → hindering automatic (near realtime) data selection and integration

- Existing techniques are not adequate for dynamic data contract exchange in data marketplaces

Need generic exchange models suitable for different ways of data provisioning in data marketplaces
Data Contract Exchange as a Service

- Metamodel for data contract exchange
  - More than data contracts themselves
- Techniques for enriching and associating data assets with contract terms
- Interaction models for data contract exchange

Hong Linh Truong, Schahram Dustdar, Joachim Götze, Tino Fleuren, Paul Müller, Salah-Eddine Tbahriti, Michael Mrissa, Chirine Ghedira: Exchanging Data Agreements in the DaaS Model. APSCC 2011: 153-160
Metamodel for data contract exchange

- Different category of agreements
  - Licensing, privacy, quality of data

- Extensions
  - Languages
  - Different types of agreements
  - Different specifications
Associating data with data contracts

### Solutions

- (a) directly inserting contracts into data assets
- (b) providing two-step access to contracts and data assets
- (c) linking data contracts to the description of DaaS
- (d) linking data contracts to the message sent by DaaS

<table>
<thead>
<tr>
<th>technical solution</th>
<th>tight-coupling</th>
<th>loose-coupling</th>
</tr>
</thead>
</table>
| structured data    | (+) ciphersing possible  
(-) requires specific client 
(-) not scalable 
(-) modifies data structure | (+) data-independent  
(-) manual access only | (-) service-specific data agreement  
(-) no enforcement possible  
(+) message-specific data agreement  
(+) ciphersing possible |
| unstructured data  | (+) data agreement enforcement possible  
(+) ciphersing possible  
(-) requires specific consumer  
(-) costly data agreement injection | (+) data-independent  
(-) manual access only | (-) service-specific data agreements  
(-) no enforcement possible  
(+) message-specific data agreement  
(+) ciphersing possible |
Possible interaction models for data enriched with data contracts
Illustrating examples – insert agreement into data asset

- A pay-per-use consumer uses dataAPI of DaaS search for data
  - The consumer pays the use APIs
  - Each call can return different types of data

Example of searching people

```json
{"results": [  
  \...
  extension: {  
    agreementReference: {  
      category: "licensing",  
      content: "http://.../DAES/da/references/retrieve/people_search—license".
    },  
  },  
  "description": "Student: spatial planning, music. Location: Vienna. Interests: Everything nerdy.",
  "location": "Vienna, Austria",  
  "time_zone": "Vienna",  
  "user_id": "REMOVED",  
  "utc_offset": "3600",  
  "name": "REMOVED",  
  "scrapped_at": 1259592694000,  
  "screen_name": "REMOVED"
}, ...
```
Illustrating examples – link agreements to geospatial data

- Domain-specific DaaS: different agreements for different data requests
  - Vector data of geographic features via Web-Feature-Service (WFS)
  - Terrain elevation data via Web-Coverage Services (WCS)

```xml
<identification>
  <agreementId>urn:de:icsyagreement:1</agreementId>
  <dataAsset>urn:de:icsy:asset.wcs:1</dataAsset>
<dataSource>
  http://gdi-arc1.gridlab.uni-kl.de/arcgis01/services/Hamburg_WFS/MapServer/WCSServer?service=WCS
</dataSource>
<dataAssetProvider>
  http://gdi-arc1.gridlab.uni-kl.de/arcgis01/services/Hamburg_WFS/MapServer/WCSServer?service=WCS
</dataAssetProvider>
<dataAssetConsumer>urn:de:icsy:j_goetze</dataAssetConsumer>
<creationDate>2011-07-04T20:11:15.029Z</creationDate>
<dataAgreementExchangeService>
  http://sod.infosys.tuwien.ac.at:7101/services/jersey/DAES
</dataAgreementExchangeService>
<agreementStatus>AGREED</agreementStatus>
</identification>
<extension>
  <agreementReference agreementSchema="urn:de:icsy:license:wcs" category="licensing">
    <content>
      http://sod.infosys.tuwien.ac.at:7101/services/jersey/DAES/da/references/retrieve/license_wcs
    </content>
  </agreementReference>
</extension>
</ns0:dataAgreement>
```
Illustrating examples – link agreements to geospatial data

Software can interpret and reason if the data can be used for specific purposes.
Illustrative examples – develop an app for policy compliance (1)

```java
public class TwitterCompatibilityApp implements DataAgreementAppInterface {
    // ...
    public String getResult () {
        return output;
    }
    public void setDataAgreements(List dataAgreements, boolean reference) {
        this.dataAgreements = dataAgreements;
        // ...
    }
    public void execute () {
        ModelManager mm = new ModelManager();
        String agreementReference1 = (String) dataAgreements
            .get(0);
        String agreementReference2 = (String) dataAgreements
            .get(1);
        OntModel ml = mm.loadFromAgreementContent(
            agreementReference1);
        OntModel m2 = mm.loadFromAgreementContent(
            agreementReference2);
        boolean valid1 = mm.validateModel(ml);
        boolean valid2 = mm.validateModel(m2);
        boolean compatible = mm.isCompatible(ml, m2);
        output = "<?xml version="1.0" encoding="UTF-8"  
            ?>" +
            "\n            // ...
    }
    public void setDescription(DataAgreementApp description) {
        // ...
```
Illustrative examples – develop an app for policy compliance (2)

Configuration

```xml
<dataAgreementApp>
  <withPlugin>RDF-Policy-Twitter-Mashup-CompApp</withPlugin>
  <withCapabilities>
    <dataAgreementAppCapability name="compatibility" value="true"/>
    <dataAgreementAppCapability name="validation" value="true"/>
  </withCapabilities>
  <worksWithSpects>
    <dataAgreementSpecification>
      <contentURI>http://sod.infosys.tuwien.ac.at:7101/services/jersey/DAES/daspects/retrieve/twitterpolicy</contentURI>
      <language>RDF</language>
    </dataAgreementSpecification>
  </worksWithSpects>
</dataAgreementApp>
```

Results

```xml
<compatibilityresult>
  <validation agreementReference="twitterpolicy1" value="true"/>
  <validation agreementReference="twitterpolicy2" value="true"/>
  <compatibility>false</compatibility>
</compatibilityresult>
```
Discussion time

HOW DOES NEAR-REALTIME DATA IMPACT ON DATA CONTRACT EXCHANGE?
Data Market without Marketplace?

Fig. 1. Schema for the atomic $S^2$aaS process of exchanging a single datum for cash using Bitcoin.


But what about data contract?

## Some data marketplaces and their features

<table>
<thead>
<tr>
<th>Products</th>
<th>Data type</th>
<th>Data source</th>
<th>Publishing/delivery</th>
<th>Cost model</th>
<th>Auto-lookup</th>
<th>Data contract</th>
<th>Payment</th>
</tr>
</thead>
<tbody>
<tr>
<td>MARSA</td>
<td>Realtime, streaming</td>
<td>IoT devices</td>
<td>MOM, IoT platform</td>
<td>Flexible</td>
<td>yes</td>
<td>yes</td>
<td>online billing</td>
</tr>
<tr>
<td>Xignite</td>
<td>Datasets, realtime</td>
<td>Range, finance</td>
<td>Files, API</td>
<td>Asset, delivery</td>
<td>Yes</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Amazon</td>
<td>Datasets</td>
<td>Range</td>
<td>Files</td>
<td>Free</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Azure</td>
<td>Datasets</td>
<td>Range</td>
<td>OData API</td>
<td>Subscription</td>
<td>N/A</td>
<td>Publisher offer terms</td>
<td>N/A</td>
</tr>
<tr>
<td>Factual</td>
<td>Datasets</td>
<td>Geography</td>
<td>Files, API</td>
<td>Free/subscription</td>
<td>Yes</td>
<td>Terms of services</td>
<td>N/A</td>
</tr>
<tr>
<td>TrimbleInsphere</td>
<td>Datasets</td>
<td>Geography</td>
<td>Files</td>
<td>per user/ device/ data</td>
<td>N/A</td>
<td>License Agreement</td>
<td>N/A</td>
</tr>
<tr>
<td>Gnip</td>
<td>Realtime, historical</td>
<td>Social network</td>
<td>API</td>
<td>N/A</td>
<td>Yes</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Sense2Web</td>
<td>Realtime, streaming</td>
<td>IoT devices</td>
<td>MOM, IoT platform</td>
<td>N/A</td>
<td>Yes</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Tien-Dung Cao, Tran-Vu Pham, Quang-Hieu Vu, Hong-Linh Truong, Duc-Hung Le, Schahram Dustdar, "MARSA: A Marketplace for Realtime Human-Sensing Data," Transactions on Internet Technology, 2016
CASE STUDY – DESIGN DATA MARKETPLACE

MARSA: A Marketplace for Realtime Human-Sensing Data
Cao, Tien-Dung ; Pham, Tran-Vu ; Vu, Quang-Hieu ; Le, Duc-Hung ; Truong, Hong-Linh ; Dustdar, Schahram
ACM Transactions on Internet Technology, 2016

http://dungcao.github.io/marsa/
Traffic problems in HoChiMinh City

- Crowded and unpredictable
- Needs a lot of data to understand traffics
- Lack infrastructures for collecting traffic information
- Common problems in developing countries

Figure sources: Internet

Cannot buy expensive traffic data collection systems!
Market-oriented View of traffic data scenarios

4000 citybus fleet, 0.25MB per day per bus (7.5MB/month/bus), 30GB for the fleet

1MB of GPS data = 20 USD cent → 6000 USD for the fleet operators

A mobile phone, like a bus, can receive 1.5 USD per month → ½ of 3G data bill
# Costs and benefits

<table>
<thead>
<tr>
<th>Parties</th>
<th>Costs of collecting raw data</th>
<th>Benefits from processed traffic data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bus, taxi and truck operators</td>
<td>GPS devices, Internet and mobile network subscription fees, acquiring and maintaining data at servers</td>
<td>Able to track status of their buses, knowledge of current traffic conditions to better provide services to commuters</td>
</tr>
<tr>
<td>Private car owners</td>
<td>GPS devices, mobile network subscription fees</td>
<td>Knowledge of current traffic conditions to better navigate in cities</td>
</tr>
<tr>
<td>Mobile device owners</td>
<td>Mobile devices (e.g. smartphones, tablets), mobile network subscription fees and device battery time</td>
<td>Knowledge of current traffic conditions to better navigate in cities</td>
</tr>
<tr>
<td>Video camera owners</td>
<td>Video cameras and network connections to video cameras</td>
<td>Selling of video data and traffic information</td>
</tr>
<tr>
<td>Data processors</td>
<td>Cost of raw data, infrastructures for collecting and processing raw data</td>
<td>Selling traffic data</td>
</tr>
<tr>
<td>Traffic data users</td>
<td>Buying traffic data</td>
<td>Knowledge of current traffic conditions to better navigate in cities</td>
</tr>
</tbody>
</table>
MARSA Design
MARSA description for human-sensing data marketplace
Quality of data has not supported yet
Implementation
Testbed
Example of bills

**Bill No.: 2015/03-5.1**
- **From date:** 2015-03-30 12:39:53  
- **To date:** 2015-03-30 18:40:57
- **Status:** Not Payment
- **Payment on DATA_SIZE (5.0 $ / 1 GB)**

**List of streams**

<table>
<thead>
<tr>
<th>No.</th>
<th>Stream UUID</th>
<th>Size</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>suuid1427702254973/sid1</td>
<td>0.219 GB</td>
<td>$1.1</td>
</tr>
<tr>
<td>2</td>
<td>suuid1427702254973/sid2</td>
<td>0.0217 GB</td>
<td>$0.11</td>
</tr>
<tr>
<td>3</td>
<td>suuid1427702254973/sid3</td>
<td>0.0550 GB</td>
<td>$0.28</td>
</tr>
<tr>
<td>4</td>
<td>suuid1427702254973/sid4</td>
<td>0.181 GB</td>
<td>$0.9</td>
</tr>
<tr>
<td>5</td>
<td>suuid1427702254973/sid5</td>
<td>0.205 GB</td>
<td>$1.02</td>
</tr>
</tbody>
</table>

**Total price:** $3.41

**Payment on SUBSCRIPTION (2.0 $ / 1 HOUR)**

**List of streams**

<table>
<thead>
<tr>
<th>No.</th>
<th>Stream UUID</th>
<th>Size</th>
<th>Price</th>
<th>Size Extra</th>
<th>Price Extra</th>
<th>Sum Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>suuid1427702254973/sid11</td>
<td>3.67 HOUR</td>
<td>$7.34</td>
<td>0</td>
<td>$0</td>
<td>$7.34</td>
</tr>
<tr>
<td>2</td>
<td>suuid1427702254973/sid12</td>
<td>6.02 HOUR</td>
<td>$12.04</td>
<td>0</td>
<td>$0</td>
<td>$12.04</td>
</tr>
</tbody>
</table>

**Total Price:** $19.38

**Total price of contract:** $22.79
Exercises

- Read mentioned papers
- Examine existing data marketplaces and write DEMODS-based specification for some of them
- Develop some specific data contracts for open government data
- Work on some algorithms for checking data contract compatibility
- Incorporate data marketplaces concepts into your scenario
- Build your own mini data marketplace
Thanks for your attention

Hong-Linh Truong
Distributed Systems Group, TU Wien
truong@dsg.tuwien.ac.at
http://dsg.tuwien.ac.at/staff/truong
@linhsolar