



# Input to ORMS TC

## Definitions and reference model proposals, Use cases

Daniela Bourges Waldegg  
[dbw@zurich.ibm.com](mailto:dbw@zurich.ibm.com)

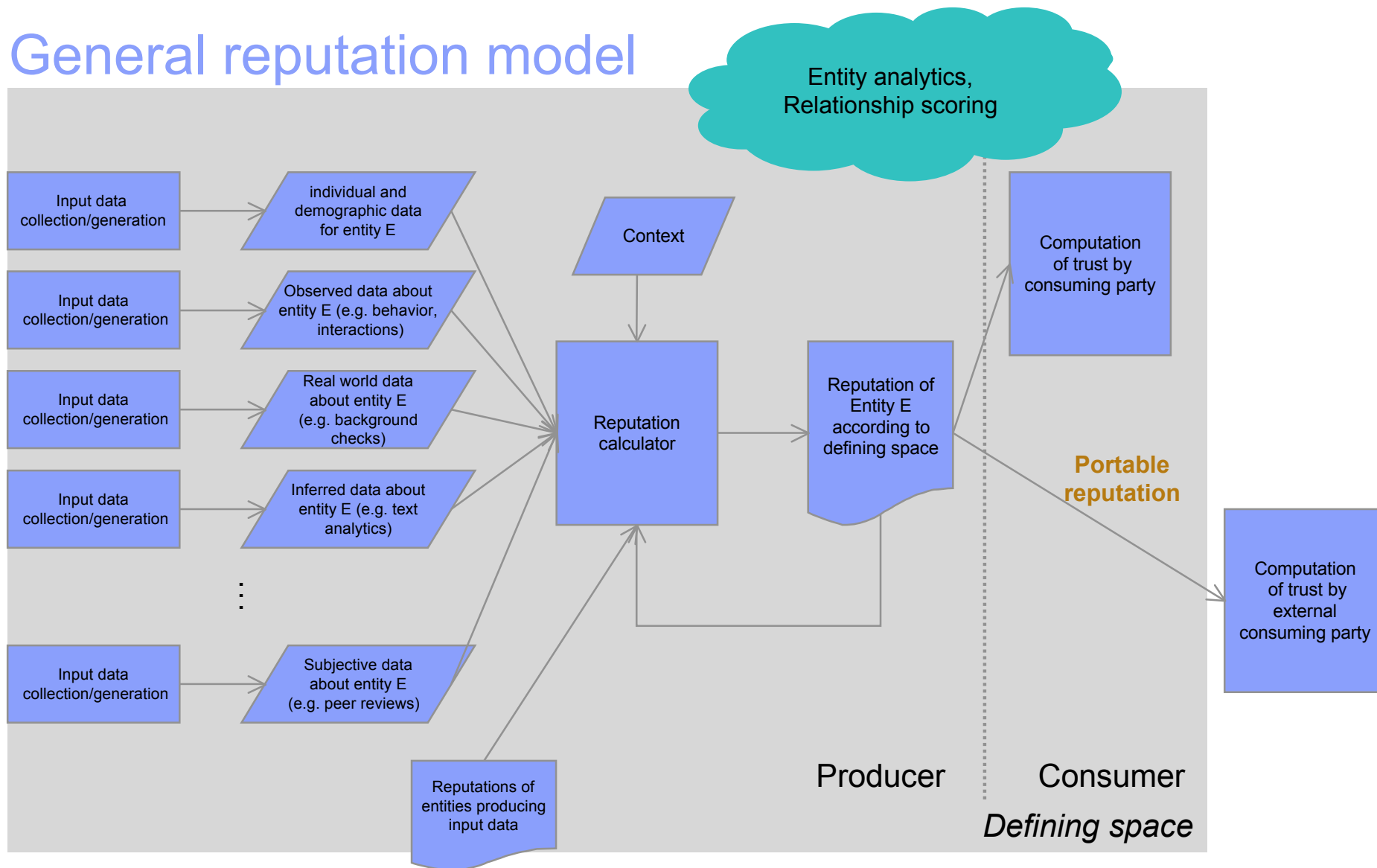
## Reputation definitions

- **Reputation is a collective evaluation of an entity based on factual and/or subjective data about it, and is used as one of the factors for establishing trust on that entity for a specific purpose**
  - Reputation can be viewed as a prediction of an outcome (an indicator that a certain outcome will occur)
- **A reputation is a metric (a score, a rank, a state, a multi-dimensional profile)**
  - associated to an entity: a person, a business, a digital identity, a website, a system, a device, a computing resource ...
  - or associated to a pair [entity, attribute] (e.g. [person, skill])
  - in a particular defining space (e.g. a community, an application)
  - at a particular moment in time (reputations are dynamic)

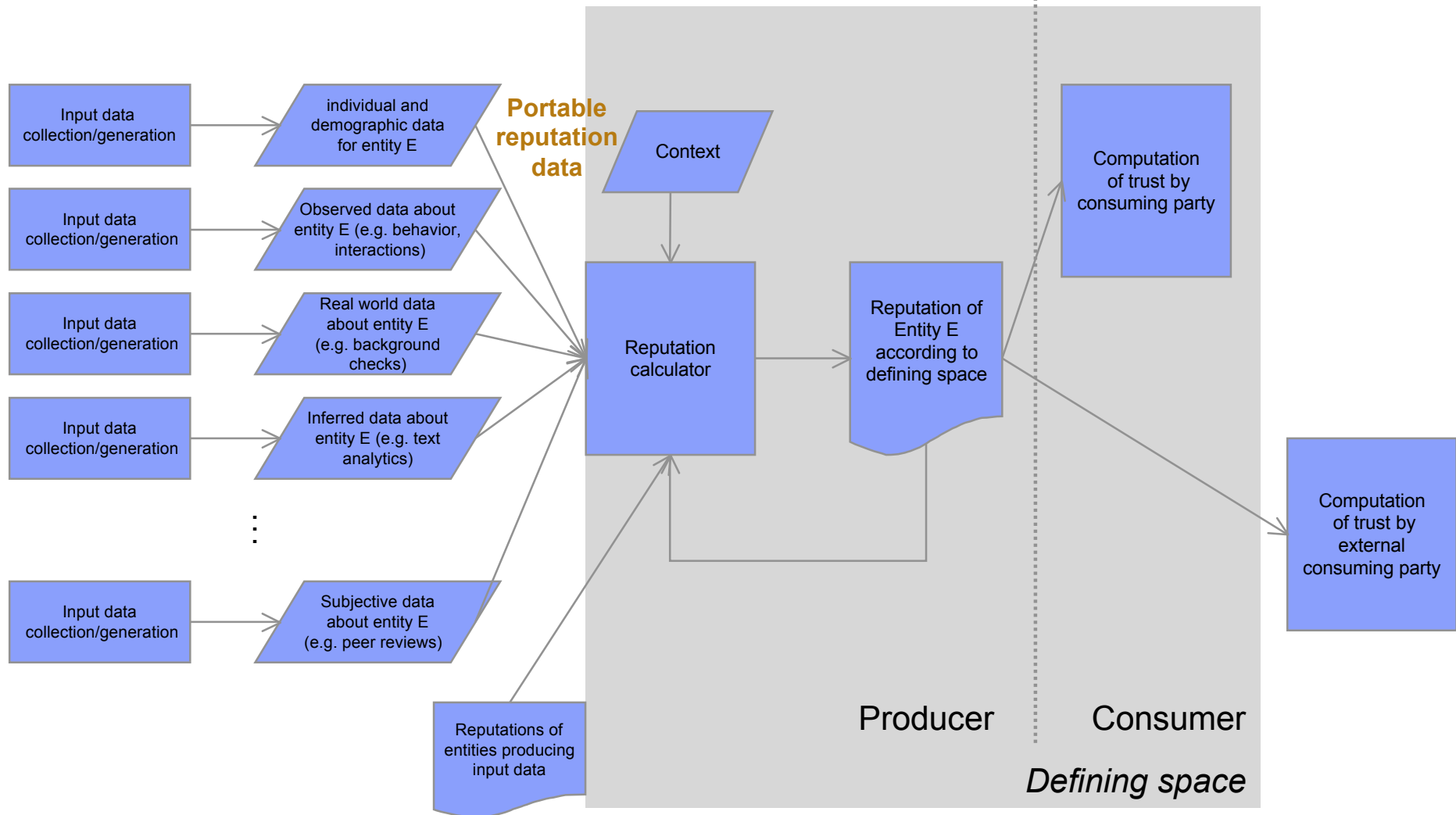
## Reputation definitions (cont'd)

- **A reputation is computed using a *reputation calculator*, based on different types of *input data* (one or several of these), for example:**
  - Static and dynamic characteristics of the entity: e.g. demographics, preferences
  - Data stemming from measurements and observations within the system: e.g. logs of entity's past behavior, history of interactions, parametric data
  - “Real world” data about the entity: e.g. background checks
  - Inferred data about an entity: e.g. text analytics
  - Subjective data associated to the entity: e.g. ratings and feedback from peers, claims and endorsements
- **The reputation calculator combines and weights one or more types of input data, according to a *reputation algorithm* and a *context***
- **Notes on input data**
  - Input data can be absolute (context-independent) or relative (context-dependent)
  - Quality and freshness of input data is key, and it should affect the way the data is weighted by the calculator
    - Reputation of the data source can also be used to infer input data quality (chained reputation)

# General reputation model



# General reputation model

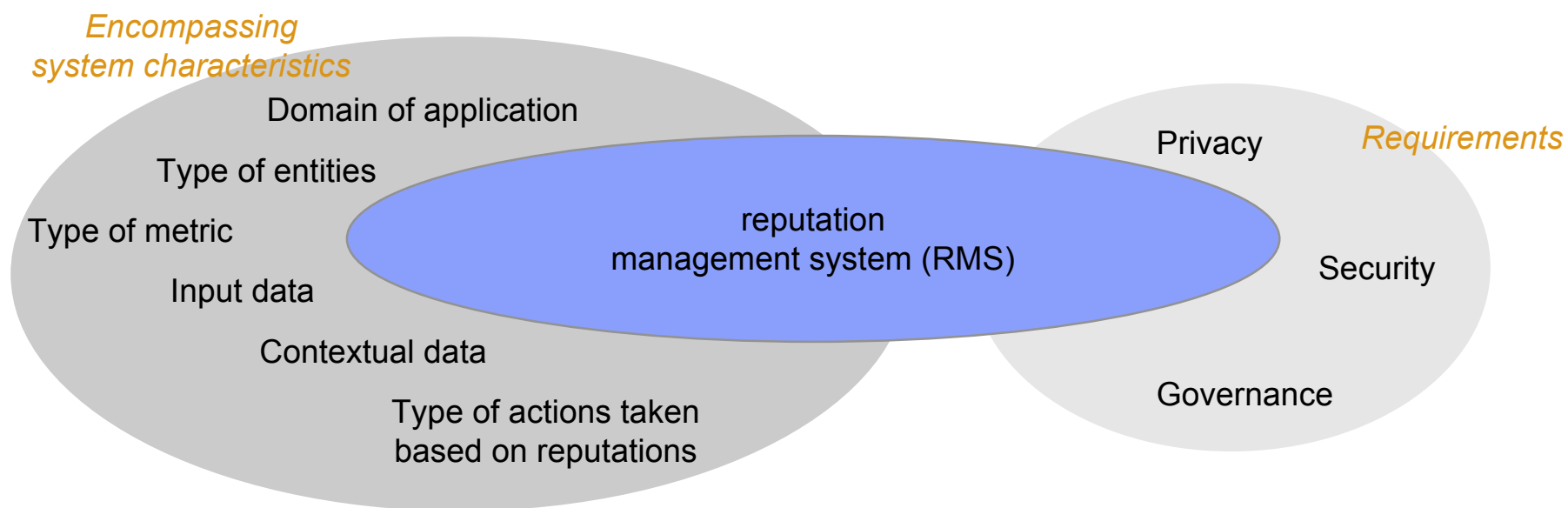


# Reputation Management System (RMS)

- **A reputation management system may include mechanisms for:**
  - Collecting data about entities (generating data inputs or integrating external data feeds)
  - Computing reputations
  - Making sure the system is fair (e.g. provide bootstrapping mechanisms for new entities)
  - Performing actions based on reputations (e.g. trust computations)
  - Revoking reputations, allowing entities legitimate control over their reputation
  - Making sure the system is not abused (e.g. by social engineering) and reputations can be challenged (to satisfy legal requirements)
  - Making sure privacy of entities is respected (i.e. that the association entity - reputation is only disclosed to authorized parties)
- **Key requirements**
  - Governance
  - Security (abuse detection/prevention)
  - Privacy: control over access to reputations, privacy preservation for sources of input data


# Characterization of reputation management systems


- **A reputation management system has many dimensions, most of which are highly system-specific**
- **Approach: look at use cases, detail some of the defining, individual elements according to reference model**
- **Guiding questions: how to design reputation algorithms? How to represent reputations and associated data?**



## Portable reputations - technologies and use cases

Area	Examples (existing technologies, possible scenarios)	Pattern
Dedicated reputation systems	IBMr	Entities = [users,skills] Input data = peer ratings
	reputation broker	
Virtual worlds	Reputations in SecondLife	Entities = avatars, objects, islands Input data = peer ratings and behavior
	Reputations for the SL Grid	Entities = avatars, SL nodes Input data = ratings of spaces, ...
Communities, collaboration spaces and social networking (Web 2.0)	Technical communities and open source development, Application reputation, Collective blogging, Open collaboration platforms	Entities = users, skills, applications, projects, ... Input data = peer ratings, history, behavior, inferred data,
	Lotus Bluehouse (collaboration platform where members are SMBs)	Entities = users, companies, ... Input data = ratings, history, behavior,...
Content filtering	Trust-based content filtering using portable reputations, Parental controls	Entities = users, systems (e.g. websites)


 Portability is intrinsic to use case


 Portability is necessary if system/community is opened



# Portable reputations - technologies and use cases (cont'd)

Area	Examples (existing technologies, possible scenarios)	Pattern
Retail and E-commerce	Buyer and seller reputations, Service and product recommendations	Entities = Users, products, services, suppliers, ... Input data = peere ratings, usage statistics, ...
Enterprise applications	Using reputation of data sources to resolve conflicting inputs in master data management	Entities = business objects, ...
Security, IT governance, systems management	Authorization/access control, IDS correlation, Information leakage detection, User & usage monitoring, device reputation, SPAM filtering based on reputation of IP addresses, Finding authoritative DNS servers	Entities = users, applications, devices and device categories,, widgets, systems, IP addresses, ... Input data = history, usage statistics, ...
“Real world” reputations	Corporate reputation analytics Reputation-aware brokerage of financial services (reputation as aid in risk calculations), using reputations in call centers, Border security Background checks for trusted identities	Entities = people, companies, brands, ... Input data = text analytics, real word metrics (e.g. credit rating),...

 Portability is intrinsic to use case

 Portability is necessary if system/community is opened



## Extra material



## Paul Resnick's classic definition (2000)

- **A reputation management system collects, distributes, and aggregates feedback about past behavior. A reputation system gives people information about others' past performance. It can enhance an on-line interaction environment by:**
  - helping people decide who to trust;
  - encouraging people to be more trustworthy;
  - discouraging those who are not trustworthy from participating.
- **Properties**
  - Entities in the system must be long-lived enough to ensure expectation of future interaction.
  - Feedback concerning current interaction is elicited and distributed and must be visible in the future.
  - Feedback must have influence on the actions/trust of entities in the future.