

# PP-PA Concepts

---

**ebXML TP F2F 10/2000**  
**Concepts/Discussion Information**  
**Scott Hinkelman**

---



# Party

---

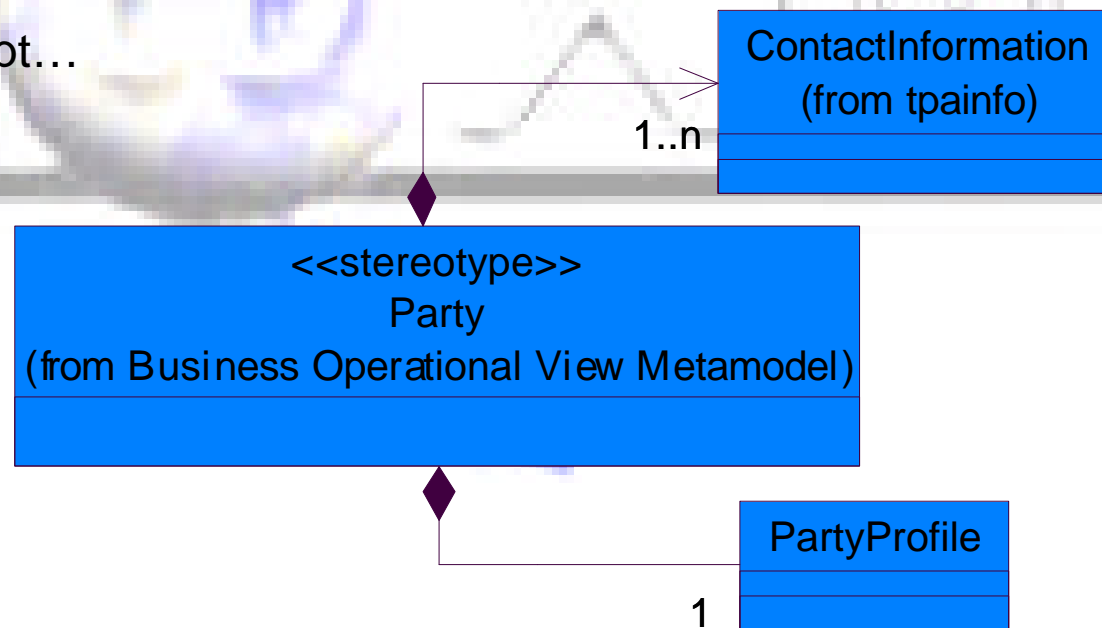
- ✗ Issue: Party or Partner? I can see Partner if we consider a wider environment scope including RA....
- ✍ Is the primary abstraction for an “involved” organization, company, individual
- ✍ Has first-order relationships to ContactInformation, PartyProfile, etc
- ✍ Contents of ContactInformation out of scope for tp because it contains no agreeables
- ✍ PP in scope for tp –contains agreeables
- ✍ Is registered within a “white page”-oriented classification in directory services (with base Contact Information, etc).
- ✍ Registers references to supporting classified BPs/Roles, and supporting technology capabilities in the form of a PartyProfile



# Party Profile

- ✍ Defines what a Party \*CAN\* do
- ✍ One PartyProfile per Party
- ✍ Party owns (lifecycle) its PartyProfile
- ✍ Dually navigable between Party and its PartyProfile

Concept...



# Party Profile....

---

- ✍ Terminology/semantics Issue:  
Roles/BusinessCollaborations/CommercialTransactions, etc. Consider ebXML Glossary here also.
- ✍ References which Roles are supported within which \*classified\* BusinessProcesses.....
  - ✍ Example:  
/Industry/Travel/Airline/FareQuoteBP/FareSupplierRole
- ✍ By directional reference between BP/Role and PartyProfile.....
  - ✍ Which BP/Roles does this Party support?
  - ✍ Which Parties support this BP/Role?

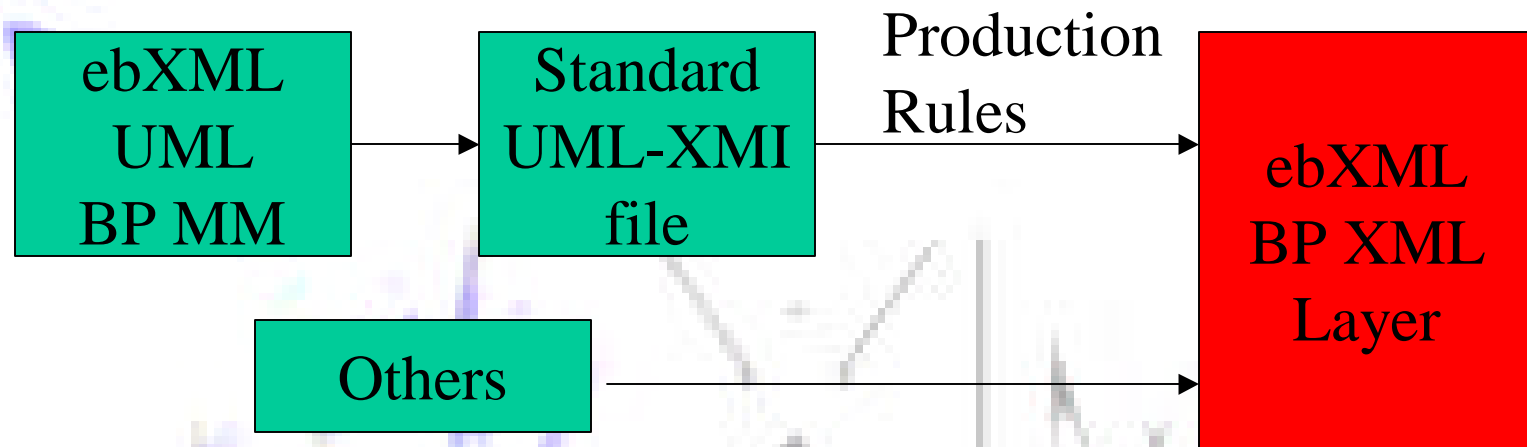
# Party Profile....

---

- ✍ Base information from a business perspective.....
  - ✍ References to BP/Roles supported
    - ✍ Implies BP/Roles defined first, then supported
- ✍ Base information from a technology perspective...
  - ✍ The underlying Transport (HTTP, FTP, etc) capability on a Role by Role basis. One Role has 1 or more TransportCapability.
  - ✍ SecurityCapabilities for each TransportCapability
  - ✍ TransportCapabilities and default parameter values are “ranked” according to most-likely-to be “dynamically agreed upon”.

None of this constitutes an Agreement

# Business Process XML Layer



- ✍ Absent of methodology used to achieve the process
- ✍ Neutral to modeling disciplines.
- ✍ Facilitate editing via lightweight tools
- ✍ Is the layer for relationships with PartyProfile and PartyAgreement

None of this constitutes an Agreement

# Business Process XML Layer....

---

- ✍ Predefined structure, flow is explicit. (Not a “dynamic conversation”)
- ✍ Is a flow of Roles
- ✍ Can specify the use of predefined structures: CCs. (this is unclear as of yet)
- ✍ Is registered in RegRep within some classification.
- ✍ BP/Roles references Parties claiming support.
- ✍ Does not require specifying technology parameters, but may include a BusinessProcessTemplate that suggests/captures reusable, incomplete Agreements that need to be parameterised.
  - ✍ tpaML has this concept but needs associated to registered/Identifiable Business Processes.

None of this constitutes an Agreement

# Party Agreement

---

- ✍ Defines what a Party \*WILL\* do
- ✍ Aggregation point for supporting the runtime
- ✍ Loosely similar to a “configuration file”
- ✍ Holds the “instance-level” data for a specific BusinessProcess runtime instance; -instance data for exactly which Parties playing which BP/Roles, over specific Transport bindings/parameters.
- ✍ Unclear on the level of CCs, but assume CC instances
- ✍ Should be designed for 2 or more Parties
- ✍ Can be context-registered in RegRep and identified
- ✍ **Issue: Discuss Lifecycle issues**

# Party Agreement....

---

- ✍ Can be constructed “manually” between Parties and distributed
- ✍ Combination of the Business BP/Roles and technology
  - ✍ Step3 of BP4 between PartyA and PartyB is agreed to be delivered via SMTP/Email
  - ✍ Step3 of BP4 between PartyA and PartyC is agreed to be delivered via HTTP
- ✍ Can be dynamically negotiated.
  - ✍ Should be thinking of “Agreeables” and which they are
  - ✍ Should design “Agreement Negotiation Protocols”
  - ✍ Bootstrap issues on how to start the negotiation
- ✍ **Issue/discussion: implied Agreements**

# Party Agreement....

---

- ✍ Can be thin and contain references to PartyProfiles, etc
- ✍ Can be fat and contain everything required for the runtime



XML

End

---

