

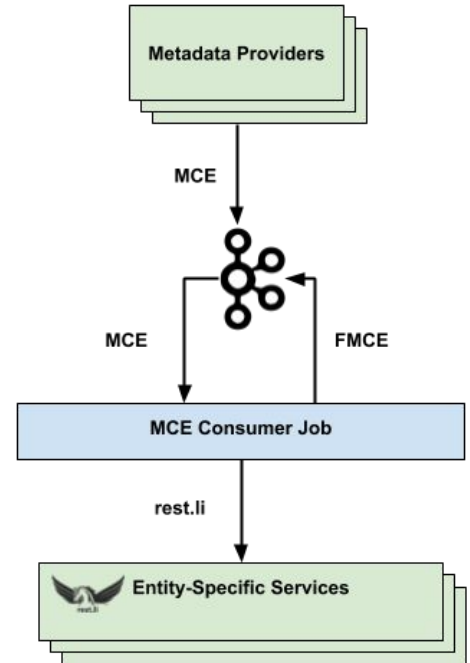


Designing the next generation of metadata events for scale

Presenter : Chris Lee
Contributors: John Plaisted, Mars Lan

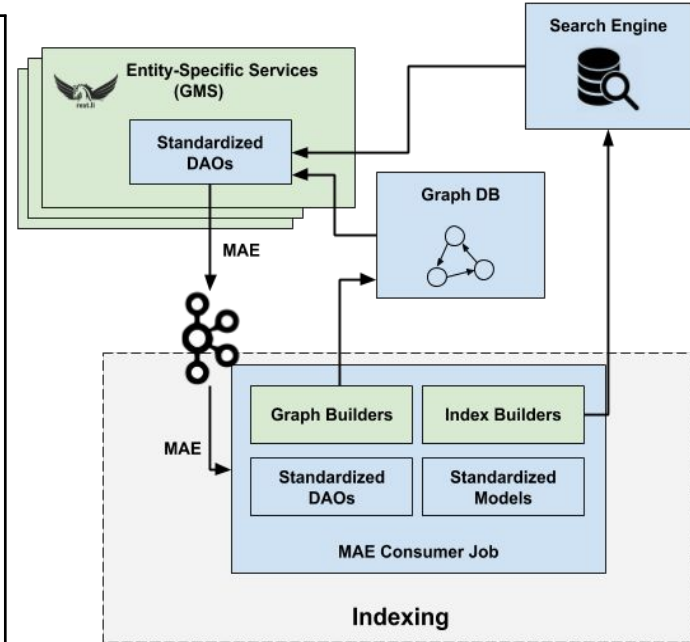
Metadata Change Event & Failed MCE

```
{
  "type": "record",
  "name": "MetadataChangeEvent",
  "namespace": "com.linkedin.mxe",
  "doc": "Kafka event for proposing a metadata change to an entity.",
  "fields": [
    {
      "name": "proposedSnapshot",
      "doc": "Snapshot of the proposed metadata change. Include only the aspects affected by the change in the snapshot.",
      "type": "com.linkedin.metadata.snapshot.Snapshot",
      "optional": true
    },
    {
      "name": "proposedDelta",
      "doc": "Delta of the proposed metadata partial update.",
      "type": "com.linkedin.metadata.delta.Delta",
      "optional": true
    }
  ]
}
```



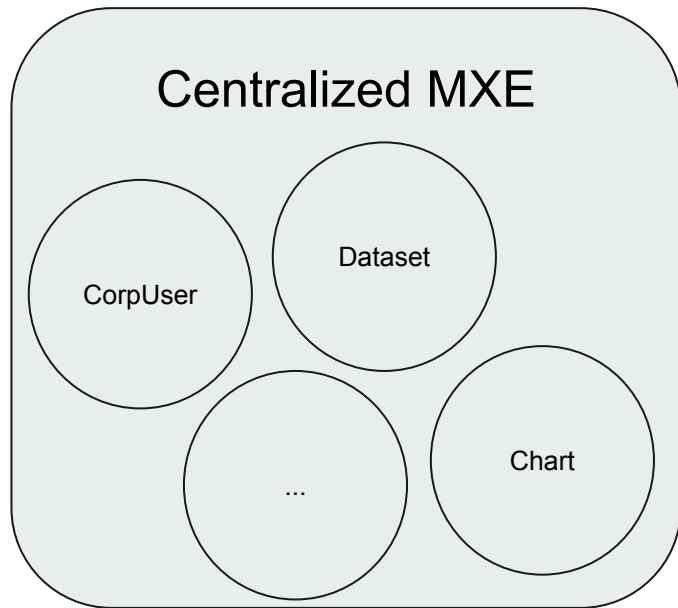
Metadata Audit Event

```
{
  "type": "record",
  "name": "MetadataAuditEvent",
  "namespace": "com.linkedin.mxe",
  "doc": "Kafka event for capturing update made to an entity's metadata.",
  "fields": [
    {
      "name": "oldSnapshot",
      "doc": "Snapshot of the metadata before the update. Set to null for newly created metadata. Only the metadata aspects affected by the update are included in the snapshot.",
      "type": "com.linkedin.metadata.snapshot.Snapshot",
      "optional": true
    },
    {
      "name": "newSnapshot",
      "doc": "Snapshot of the metadata after the update. Only the metadata aspects affected by the update are included in the snapshot.",
      "type": "com.linkedin.metadata.snapshot.Snapshot"
    }
  ]
}
```





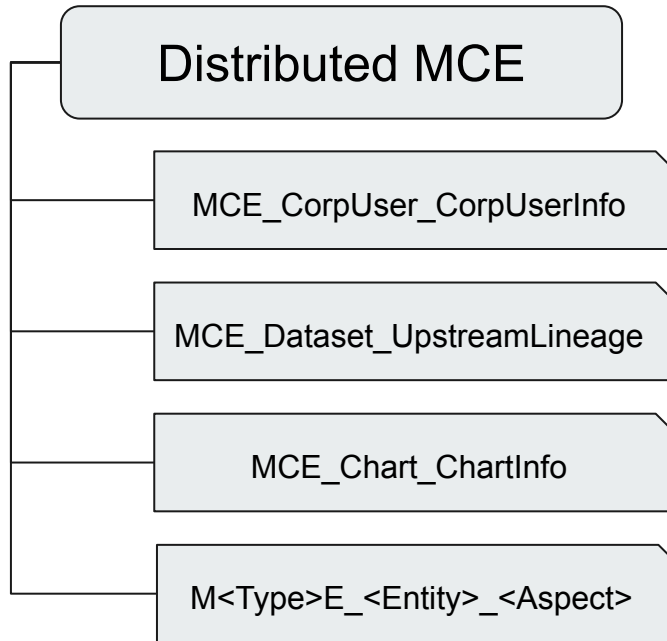
Motivation



- Global impact on backward incompatible changes.
- One topic ACL for all entities.
- Potentially hit the schema registry limitation.
- Less offline system support.



Aspect-Specific MXE



- Local impact on backward incompatible changes.
- Aspect fined level ACLs for each topics.
- Less likely to hit the schema limitation.
- Straight forward query in offline system.



PDL annotation

```
@gma.aspect.entity.urn =  
"com.linkedin.common.urn.CorpUserUrn"  
record CorpUserInfo {  
  // ...  
}
```

auto-generated



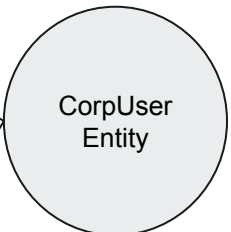
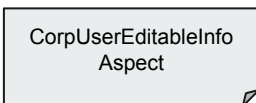
```
record MetadataChangeEvent {  
  // ...  
  urn: CorpuserUrn  
  // ...  
  proposedValue: optional CorpUserInfo  
}
```

```
record FailedMetadataChangeEvent {  
  // ...  
  metadataChangeEvent: MetadataChangeEvent  
  // ...  
  error: string  
}
```

```
record MetadataAuditEvent {  
  // ...  
  urn: CorpuserUrn  
  // ...  
  oldValue: optional CorpUserInfo  
  // ...  
  newValue: CorpUserInfo  
}
```

Onboarding workflow

Annotate w/urn



Annotate w/urn



Generate PDL schema



Generate Avro schema



Create Kafka topic & register schema



Setup ACL for producers & consumers

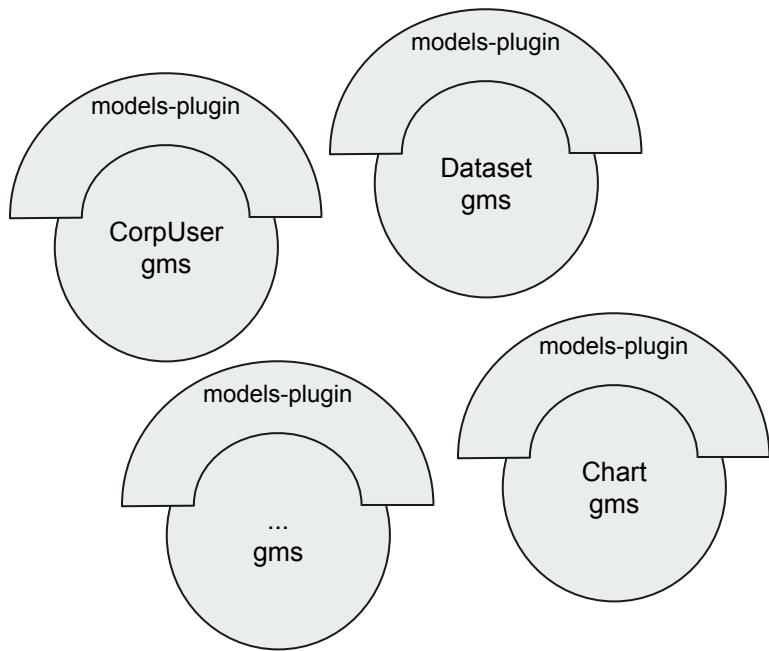


Developer action
Automated flow

M<X>E_CorpUser_CorpUserInfo &
M<X>E_CorpUser_CorpUserEditableInfo



Future Work



- Enable distributed GMSes
 - Apply model code-gen plugin.
 - Aspect-specific models live in its gms.
 - Common models stay in central repo for reference.
- Extend annotation properties
 - Open source oriented attributes.
 - Uri template in request builders.



Q & A

Recommendations & Suggestions are welcome 😊