

IPU Consumption Best Practices

Sriram Seshadri
Principal Solutions Architect

Agenda

- IPU Recap
- CDI/CDQ IPU usage Best Practices
- Cloud App Integration IPU usage Best Practices

The New Informatica Processing Unit (IPU)

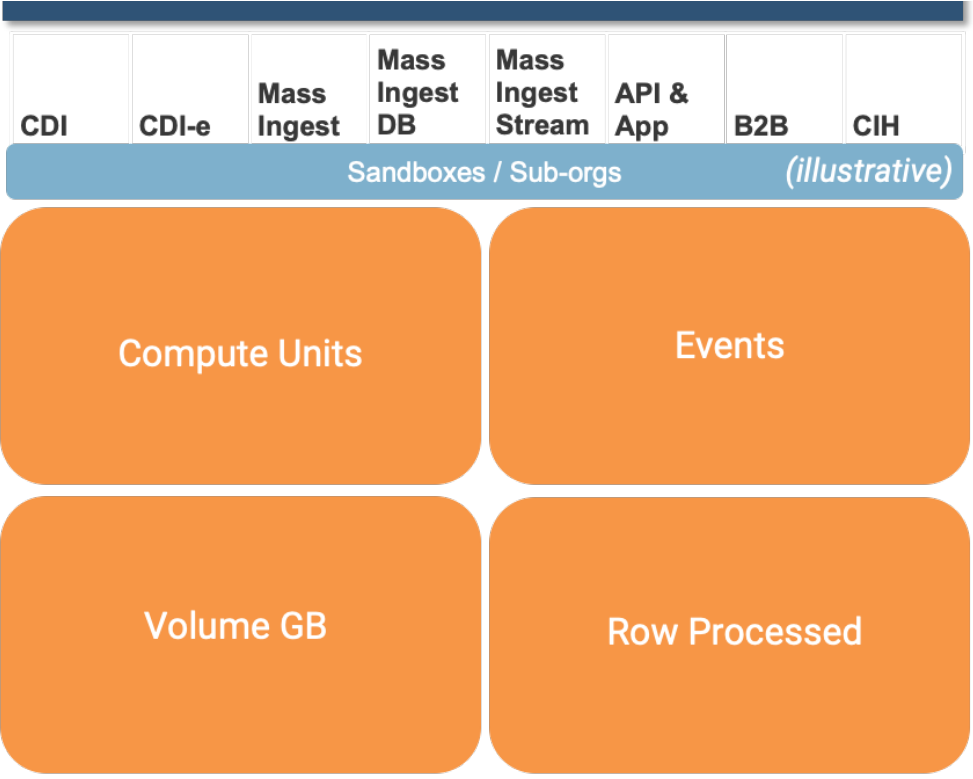
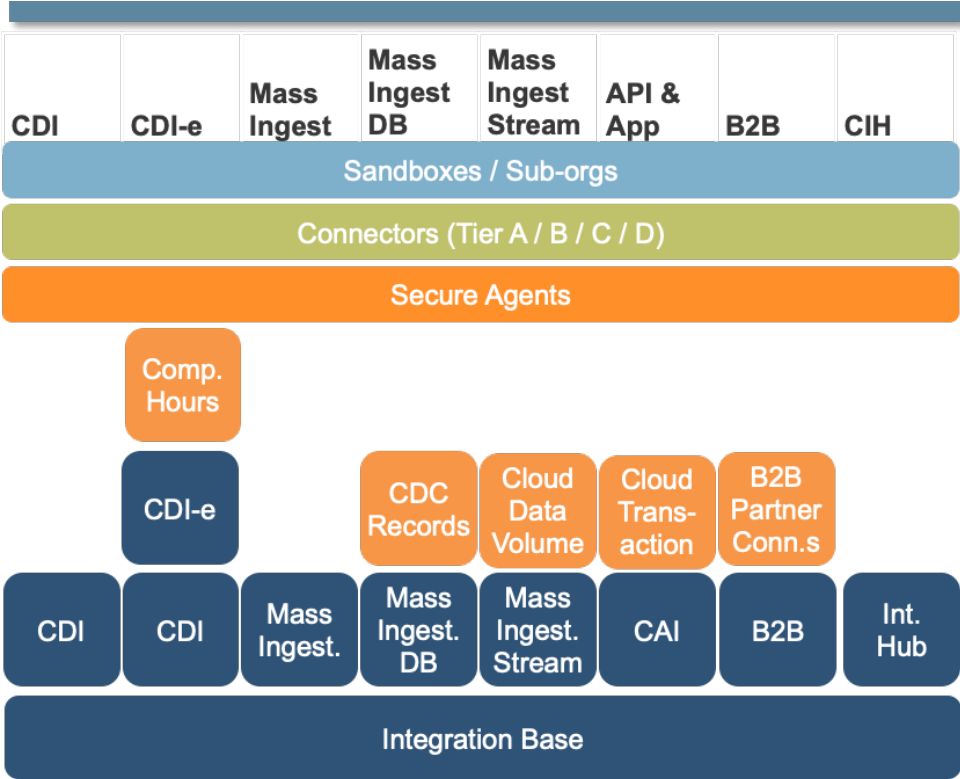
Unit of software licensing capacity that can be used across a **range of cloud services**

Can be **interchangeably** used across different cloud integration and quality services

Customers purchase IPU **on a pre-paid subscription basis** and **use any service** up to the pre-paid IPU subscription capacity

The new IPU model offers more value and benefit by granting access to a wider range of services and measuring their use on a consumption basis

Old vs New Packaging



Cloud Data Integration

- Meter calculated based on cores used and duration of job.
- Use Detailed report to analyze metered hours and duration.
- Use Advanced PDO if compatible to offload data processing away from agent server.
- Schedule Conflicts. There can be too many jobs part of the same schedule
 - Resolution: If it can be accommodated, spread the schedules to limit resource usage. Disable if many jobs are running on schedule on dev or other sandboxes(possibly turned on for testing) which can be turned off
- Split CDQ jobs from CDI processing if possible as CDQ meter is higher.
- Resource Crunch (low CPU and memory) which slows down the job waiting for resources.
 - Resolution: Use Ops Insights to view usage and size up accordingly. Set alerts for threshold breach. Check if maxDTMProcess is higher than server can handle. Allocate sufficient memory to the agent and JVM heap.
- Long Duration Jobs. Jobs taking longer than average runtime.
 - Resolution: Use ops insights to be alerted of breaches and kill task (if possible) to avoid IPU leak

Cloud Data Integration(Contd)

- Use the right pattern based on use case. For eg, trying to create ingestion use cases using CDI increasing duration and cores used. CMI meter might be efficient as the meter goes by volume/rows
- Deep dive into logs to see which transformation spends a lot of time. For eg, you could have lookups against a large table causing slowness in prep stage.
- Source query time being extremely slow. For eg, reading 10K records from a source takes 1 hour. Try to optimize working with the team managing the source/DB. This can also be assessed looking at session logs.

Cloud App Integration

- Meter calculated based on Process Runtime.
- By default, the process runs on the cloud server. Lot of times, the process needn't be exposed to external partners and be available over the internet.
 - Resolution: Deploy the process on the agent to avoid the IPU rate difference between running on cloud or agent.
- Add error handling or response time checks for service calls. Longer the response time from the external API, higher the IPU cost.
- CAI Process triggered for bulk load scenarios from calling client
 - Resolution: Plan the operation to be handled as a one-time batch to avoid large volume of calls(causing possible resource crunch) and increase in IPU

Thank You!