

Lou Loizides – инженер с более чем 10-летним опытом работы по реализации концепции Industry 4.0. В настоящее время он в качестве эксперта по производству занимается в компании Foghorn Systems, Inc. инновационными приложениями так называемой граничной аналитики (Edge Analytics). Эта концепция подразумевает, что сбор, обработка и анализ данных происходят на периферийных устройствах сети, рядом с датчиками, сетевыми коммутаторами или другими подключенными устройствами – то есть рядом с источником информации и исполнительными устройствами.

Оригинал статьи “*Designing A Data-Centric Industry 4.0 System*” на английском языке можно найти на сайте www.machinedesign.com.

Проектирование ориентированных на обработку данных систем для Industry 4.0

Lou Loizides

©2019 Penton Media Group



Четвертая промышленная революция изменит способ производства товаров. Такие концепции, как упреждающее техническое обслуживание, прогнозирование спроса и цифровые двойники не только уменьшают продолжительность простоя и колебания качества, но еще и могут помочь повысить эффективность производства.

Награда за приложенные усилия будет достойной. Аналитики *Price Waterhouse Cooper* ожидают, что инициативы *Industry 4.0* обеспечат рост эффективности, характеризуемый цифрой 400 млрд. долларов.

Развитие концепции *Industry 4.0* охватывает множество передовых технологий. Формируется новый, сложный и интересный пейзаж. В результате даже крупнейшие компании из списка *Fortune 500* могут столкнуться с проблемой разработки стандарта архитектуры для систем накопления и обработки потоков производственных данных.

Подходящей для всех стратегии разработки архитектуры таких систем не существует. Критически важное значение имеют классификация данных и проектирование систем, способных обрабатывать данные различных типов. Воплощение концепции *Industry 4.0* в жизнь открывает ворота большому количеству разнообразных данных, и не все они образуются одинаковым образом.

Для понимания этого следует начать с осознания того, что промышленные данные в своём большинстве являются потоковыми и привязанными ко времени. Например, вакуумная печь может генерировать замер характеристик вакуума каждую секунду, а производственная исполнительная система (*Manufacturing Execution System, MES*) может генерировать данные о выполнении операции каждые пять минут. Хотя и содержание, и форма этих информационных элементов отличаются, общим является то, что они формируют потоки.

Потоковые данные можно классифицировать по таким характеристикам, как требуемая пропускная

способность канала, способы использования [и др.] При группировании промышленных данных по этим атрибутам вырисовываются следующие три логических класса данных: граничные, локальные и используемые удаленно (табл. 1).

Граничные данные

Граничные данные образуются на самой границе сети. Они поступают с высокой скоростью в режиме реального времени и не отправляются на хранение. Вышеупомянутая вакуумная печь может генерировать данные о вакууме и температуре с очень высокой скоростью.

До эпохи *Industry 4.0* сгенерированные данные прореживались, и, таким образом, большая их часть отбрасывалась. Потоки данных, которые не дают инженерной отдачи, были бы отброшены полностью.

Проблема с отбрасыванием этих данных, конечно же, заключается в том, что теряется огромное количество потенциально ценных сведений. К примеру, представим, что у печи возникли проблемы с вакуумным насосом. Если бы собираемые с высокой частотой сканирования данные и машинная логика были сохранены, то можно было бы выявить шаблон или отличительное свойство для диагностики этого состояния и сформировать запрос на обслуживание в режиме реального времени.

Кроме того, обработка комбинаций потоков данных с датчиков (объединение датчиков), может дать очень ценную информацию о производственных операциях. Представьте себе, что детали, входящие в печь, имеют видимые штрих-коды. Совместное использование сканера штрих-кодов, логики системы управления и рабочих характеристик печи дает оператору полную прозрачность в отношении статуса каждой детали в процессе. Комбинируя этот контекст с текущими характеристиками процесса и прогнозом готовности, можно строить прогнозные модели в режиме реального времени, что поможет повысить как эффективность использования машин, так и качество продукции.

Проблемой с применением граничных данных, конечно, является пропускная способность канала.

Хранить информацию от имеющихся на производстве десятков или даже сотен тысяч датчиков и все логические потоки – это дорого и непрактично. Единственный способ высвободить заключенный в ней смысл – разместить высокоскоростные аналитические приложения рядом с источниками генерируемых данных.

Способная приносить пользу платформа граничной аналитики соединяет в себе инструменты обработки данных в режиме реального времени, визуализации в режиме реального времени и расширенные возможности машинного обучения. Программные движки потоковой аналитики обычно не могут создаваться с помощью традиционных языков программирования – из-за событийной природы потоковых данных от сенсоров. Каждая платформа имеет свое собственное уникальное решение для разработки приложений потоковой аналитики. Эти движки характеризуются разнообразием, но, чаще всего, они основываются на сложной обработке событий (*Complex Event Processing, CEP*) или на правилах.

Так как индивидуально управлять сотнями или тысячами границ непрактично, то граничная аналитика должна сочетаться с централизованным управлением – для координации распределения работ и удаленного выполнения. Извлеченная средствами потоковой аналитики полезная информация, которая относится к более высокому уровню

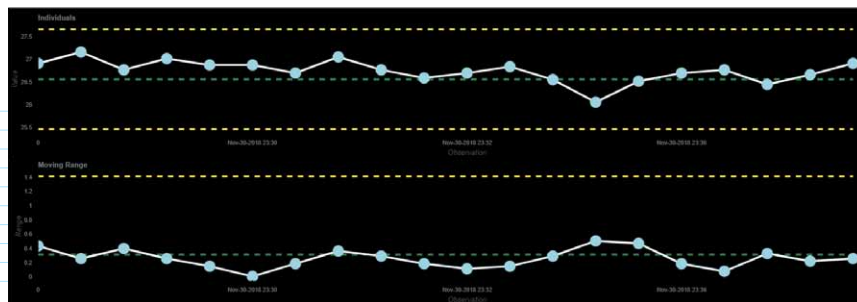


График статического управления технологическим процессом в режиме реального времени, построенный CEP-движком по данным, полученным от различных сенсоров

(локальное или удаленное использование), может далее передаваться в любую другую информационную систему или базу данных.

К недостаткам предлагаемых граничных платформ, на которые следует обращать внимание, относятся: ограниченная функциональность, сложность использования конфигураций и проприетарные средства аналитики, которые нельзя переместить за пределы экосистемы провайдера.

Допустимы и компромиссы в виде внедрения нескольких систем потоковой аналитики. Событийные (*CEP*) движки – более гибкие и могут моделировать более сложные процессы, но они требуют больше предварительной работы по конфигурированию. Движки, основанные на правилах, имеют тенденцию ограничиваться простым мониторингом условий и готовности, но они запускаются намного быстрее и легче.

Табл. 1. Классификация потоковых промышленных данных

Характеристика	Граничные данные	Локальные данные	Удаленно используемые данные
Скорость формирования	Очень высокая (мсек)	Высокая (сек)	Медленная (мин)
Требуемая пропускная способность каналов	Высокая	Средняя	Низкая
Пример данных	Машинная логика, необработанные данные сенсоров	Прореженные данные сенсоров, двоичные датчики (например, ближней локации)	Операции с материалами, завершение маршрутов, выборочные данные о качестве, общая производительность оборудования, запросы на обслуживание
Хранение	Временное (в памяти)	БД для временных рядов (<i>TSDB</i>)	Публичное/приватное облако или озеро данных
Доступ	Нет прямого доступа – только аналитика и визуализация	Завод и инженерные отделы, связанные с ним	Вся географически распределенная компания
Визуализация	Живые графики и панели на основе архитектуры <i>Pub-Sub</i> с шиной событий	Визуализация, управляемая <i>TSDB</i>	Облачные инструменты создания панелей и мобильные приложения
Анализ данных	Потоковая аналитика и машинное обучение	Легкая локальная аналитика и машинное обучение	Инструменты для работы с большими данными

Локальные данные

На протяжении десятилетий объектом систем диспетчерского контроля и сбора данных (*Supervisory Control And Data Acquisition, SCADA*) являются локальные данные. В отличие от граничных, локальные данные не являются временными. Но для того, чтобы уменьшить требования к сети (ограничить пропускную способность гигабайтами в секунду) и хранилищам, частота выборки обычно понижается.

Хранят такие данные, как правило, тоже ближе к месту их возникновения. Эти данные полезны для информационных панелей, для отладки процессов и для использования в ходе процессов, не критичных в отношении режима реального времени. К примеру, прореженные данные о температуре и давлении в вакуумной печи могут классифицироваться как локальные и сохраняться на случай выявления глубинных проблем.

Многие граничные платформы включают в себя собственные базы данных для временных рядов (*Time Series Database, TSDB*). Отличие от реляционных БД заключается в том, что они оптимизированы для хранения и извлечения непериодических серий данных, привязанных к моментам времени. Связь между разными сериями данных обычно определяется временной меткой. Помимо сохранения критически важных данных, такая БД может использоваться для хранения и управления информационными панелями, или же для управления локальной аналитикой и состоянием готовности. Во многих популярных *TSDB* политики хранения могут устанавливаться так, чтобы данные сохранялись только определенный период времени.

Удаленно используемые данные (облако/озеро данных)

Основное отличие данных удаленного использования заключается не только в частоте выборки, но еще и в самой природе данных. Отправка [больших объемов] данных в облако и их хранение там, или даже в “озере данных” организации, как правило, обходится очень дорого и нагружает каналы связи. Поэтому хранение в любой

удаленной системе должно ограничиваться лишь теми данными, доступ к которым необходим большому количеству пользователей, разбросанных географически. Пропускная способность сети для таких данных должна быть на порядки меньше, чем это требуется для граничных данных – здесь речь идет о мегабайтах в секунду.

Таким образом, для удаленного хранения, в основном, подходят данные об операциях с деталями и о состоянии процессов. В качестве примеров типов данных, для которых целесообразно использовать централизованное озеро данных или облако, можно назвать статус процесса, завершение маршрутов, характеристики качества.

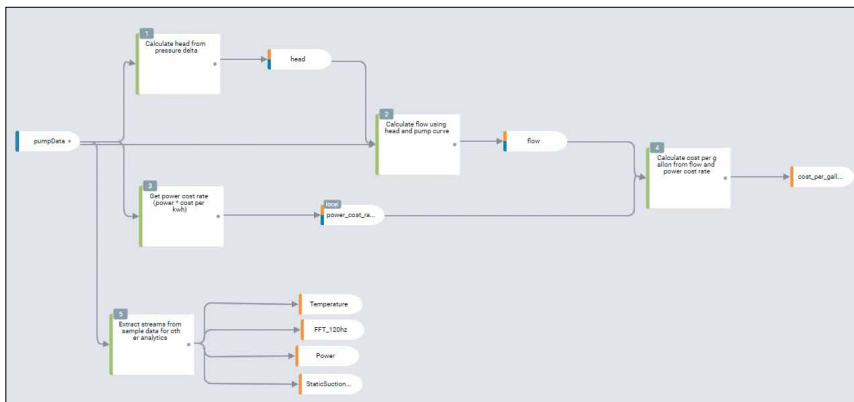
Часть объединенной информации с датчиков, выделенная из необработанных граничных данных при помощи платформы потоковой аналитики, попадает в класс удаленных/облачных данных. Вернемся к примеру с вакуумной печью, и рассмотрим возможность фиксации точного состояния (статуса) любой детали или процесса. Программный движок потоковой аналитики может перекомпоновать информацию об этих состояниях, запустить *MES* и материальные транзакции, а также извлечь любые связанные с ними характеристики качества. Данные об операциях и характеристики качества могут быть полезны для других предприятий, а также для управления цепочкой поставщиков, и они хорошо вписываются в класс удаленно используемых данных.

Заключение

Эффективная архитектура систем для *Industry 4.0* должна охватывать три типа данных – граничные, локальные и используемые удаленно. Проектирование лишь для одного типа данных подвергает инженеров риску потерять критически важную информацию о процессе и ценную возможность глубокого понимания.

Граничная обработка должна проводиться как можно ближе к источнику данных. На промышленном предприятии с надежной сетью, способной пропускать терабайты потоковых данных, приемлемым решением может быть один граничный стек. Но если скорость всего процесса работы с данными имеет решающее значение (например, для высокоскоростных данных о вибрации или когда граничные устройства используются для принятия управляющего решения или воздействия), то граничная аналитика должна находиться как можно ближе к объекту.

Для любой эффективной архитектуры, поддерживающей концепцию *Industry 4.0*, необходимо граничное решение в сочетании с архитектурой и стратегией использования всех трех классов данных. 🤖



Потоковая аналитическая логика (каждая строка – это поток)