



OMNICAST

City wide surveillance

Hiroshi Sasaki, CEO | Javatel

# Genetec- 企業概要

- 企業情報

- 個人所有
- 本社はカナダ、ケベック州、モントリオール
- 1997に創業
- 従業員210名 (80%はエンジニア)

- 中核技術

- IPセキュリティソリューションにおけるソフトウェア開発

- 現在のマーケット

- 自社ブランドのプロモーティング
- 拠点: 日本、北アメリカ、EMEA、西インド諸島(カリブ)



# Genetec

- ワールドワイドのオープン・アーキテクチャVMSマーケットで20 - 25 % のシェア
- 現在最大のシステムは200,000カメラ。今後 250,000になる予定(1システム)
- G7のうち5カ国の元首を保護
- 以下に強力です:
  - 空港
  - 公共インフラ
  - 市規模の監視



# Javatel- 企業概要



- 企業情報

- 個人所有
- 本社は大阪市、北区天神橋
- 1999に創業
- 従業員6名

- 中核事業

- IPセキュリティソリューションにおけるソフトウェア、ハードウェア開発、販売、サポート



- 現在のマーケット

- Genetec製品のプロモーティング
- 拠点: 大阪、東京 代理店 NTTドコモ、加賀ソルネット、CBC、他
- 日本のVMSマーケットで8%のシェア (テクノシステムリサーチ調べ)

# Rome disaster prevention center

ローマ災害予防センター



# OEMC control room

OEMC コントロールルーム



# West Edmonton Mall

ウェストエドモントンモール



# Greater Manchester Police

グレーター・マンチェスター警察



# 企業規模のビデオ管理

## アーキテクチャ

- 集約化された設定
  - クライアントサーバ
  - 階層化された役割を基にしたセキュリティ
- 企業都合の良いシステム管理
  - ソフトウェア開発
  - スクリプティング
  - データベース、バックアップ、仮想化標準
  - ライセンス供与
- 安価なフェイルオーバー
- 独立したネットワークポロジ
- オーナー権の全体のコスト

## 機能

- 直感的なユーザインターフェイス
  - 検出可能
  - 適度なトレーニング
  - シンプル
- マルチキャスト
- イベントとアラーム管理
- レポートエンジン
- カスタマイズ性
  - プラグイン
  - インターフェイス
  - スクリプティング
  - SDK
- 企業規模レベルのパフォーマンス

# イベントとアラーム管理



何が起きたのかを確実に知ること。

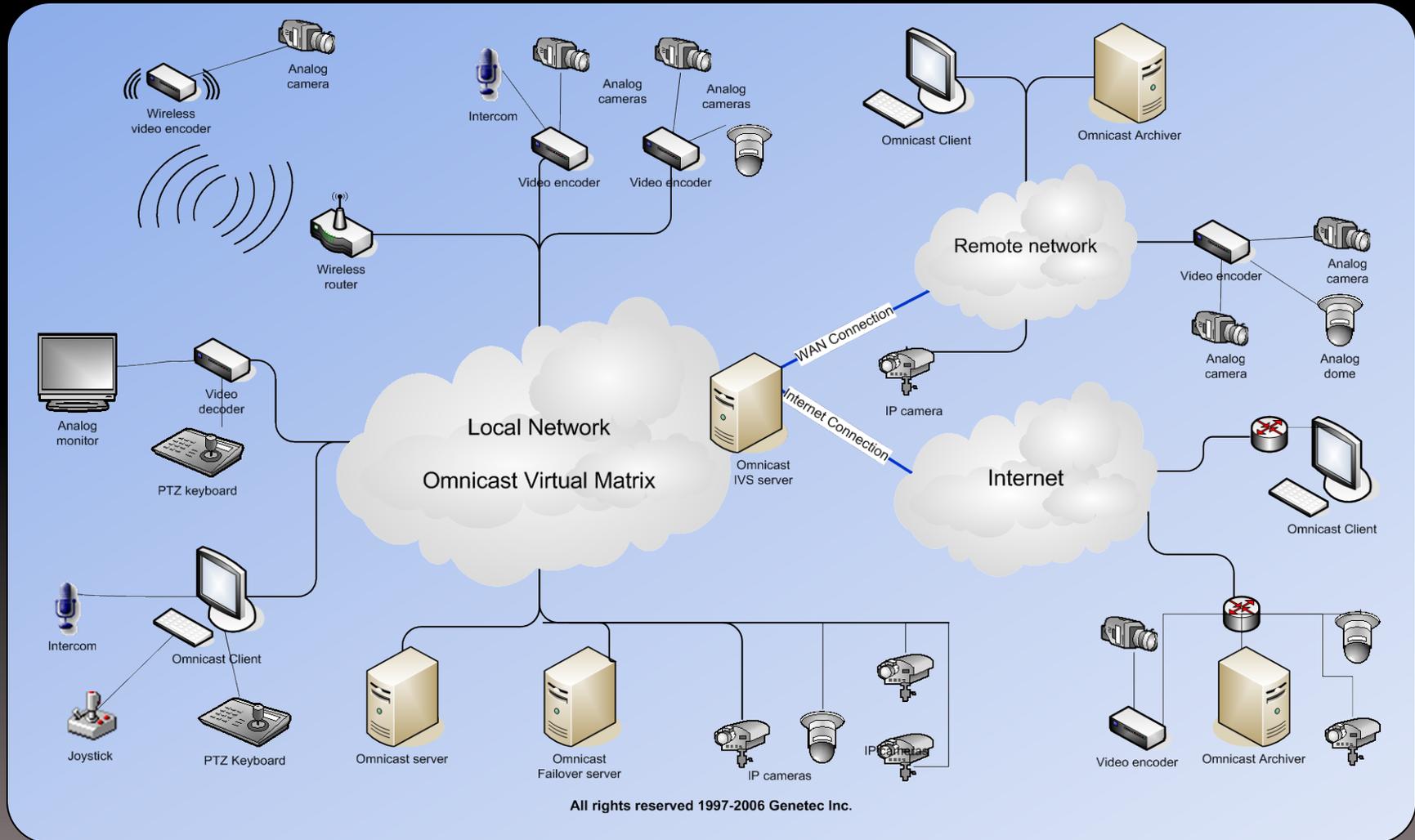
何らかの異常を検知したカメラの映像が切り替わった、しかしそこには誰もいなかった。一人の警備員は現場に向い、もう一人が手間取りながら映像を再生して状況確認をしていた。現場に駆けつけた警備員は何者かに襲撃され何も分からずその場に倒れた。再生映像を見ていた警備員は慌てて無線で呼び出したが、とき既に遅しであった。

重要な事は何が起きたのか、どのようにして起きたのかを瞬時に知ることなのです。OMNICASTの強力なアラーム管理ならイベントタイミングで最大±300秒(任意設定可)を瞬時に再生し警告します。





# Omnicast アーキテクチャ



# Our mantra

信念

拡張性

- 上、下、そして外へ
- フェデレーティング(共有)

メタデータ

- 解析
- インテグレーション
- ワークフロー

オープンアーキテクチャ

- インターフェイス
- インテグレーション
- マルチベンダー

# Scale, Metadata, Open Architecture

拡張性、メタデータ、オープンアーキテクチャ

- インターネットでの復元力
- 規模拡大化は必然、抵抗は無意味
- 共有とQOS 期待値
- 富の新しい源としての生産性
- 統合的な知覚
- ソフトウェアは永遠
  - ハードウェアは短命
  - ソフトウェアがハードウェアに対する投資を保守

# Key Feature: Federation

- 市規模の監視が可能
- 店舗/支店のリモートモニタリング

調査員



調査員は同時に複数のOmnicastディレクトリ(ローカル & リモート)へ異なった設定の資格で接続できます

資格によって、調査員は以下が行えます:

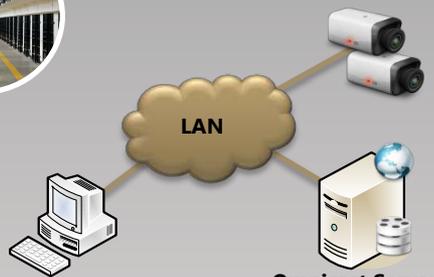
- ライブ/プレイバックビデオを見る
- PTZのコントロール
- 音声を聞く
- イベントとめたデータの管理



モニタリング  
コントロール

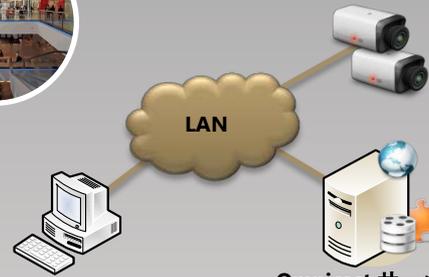
システムアクセス  
アドミニストレーション

交通局



Omnicast Server

学校

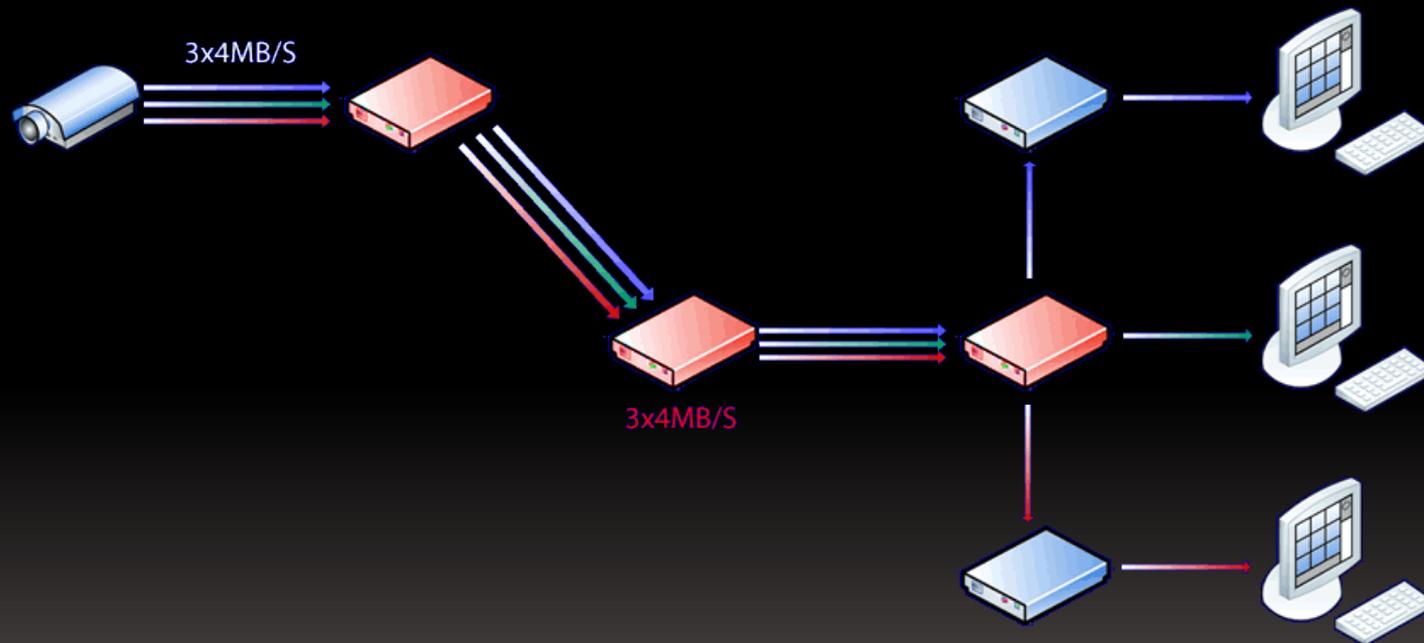


Omnicast サーバ

学校管理者が  
Federationユーザーと  
何を共有すべきかを  
決めます

# Key Feature

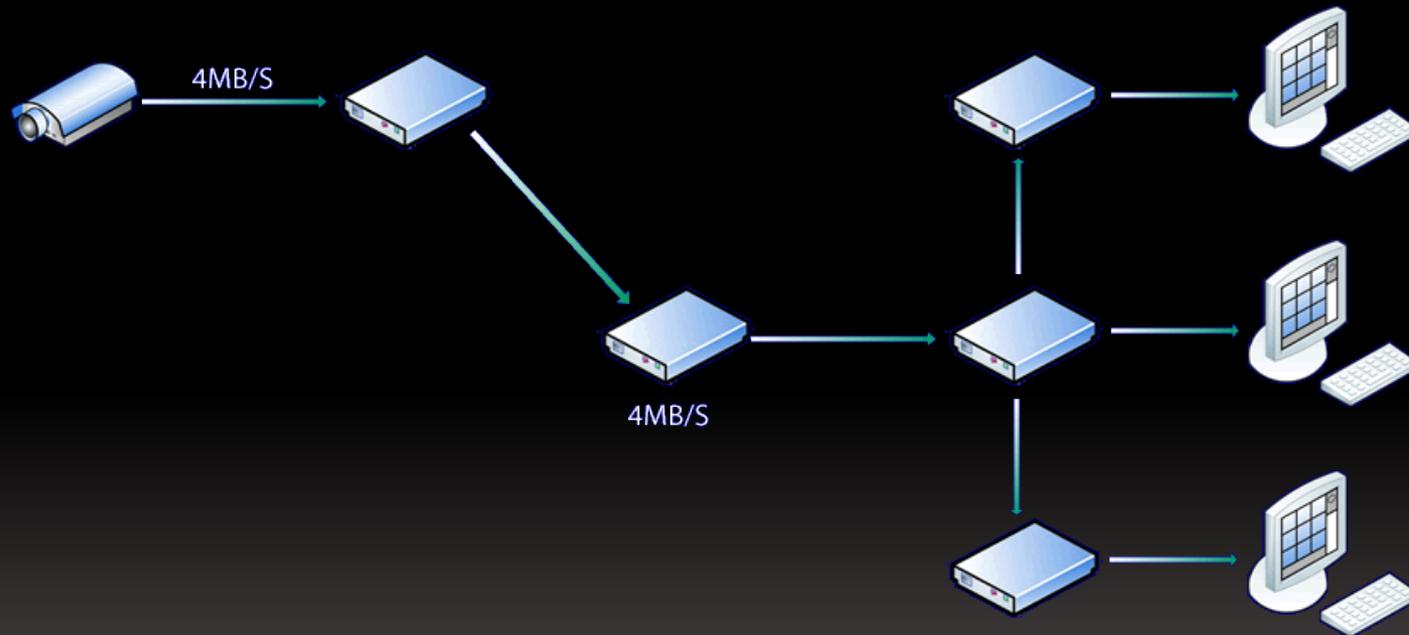
## マルチキャスト



ユニキャストでは、ネットワークコンポーネントはより重い負荷に直面します。この場合、たった1つのカメラのデータに対し3倍のデータ量になります。

# Key Feature

## マルチキャスト



マルチキャストでは、すべてのワークステーションへ1つのストリームで送られます。これによりネットワークの負荷効率を上げます。

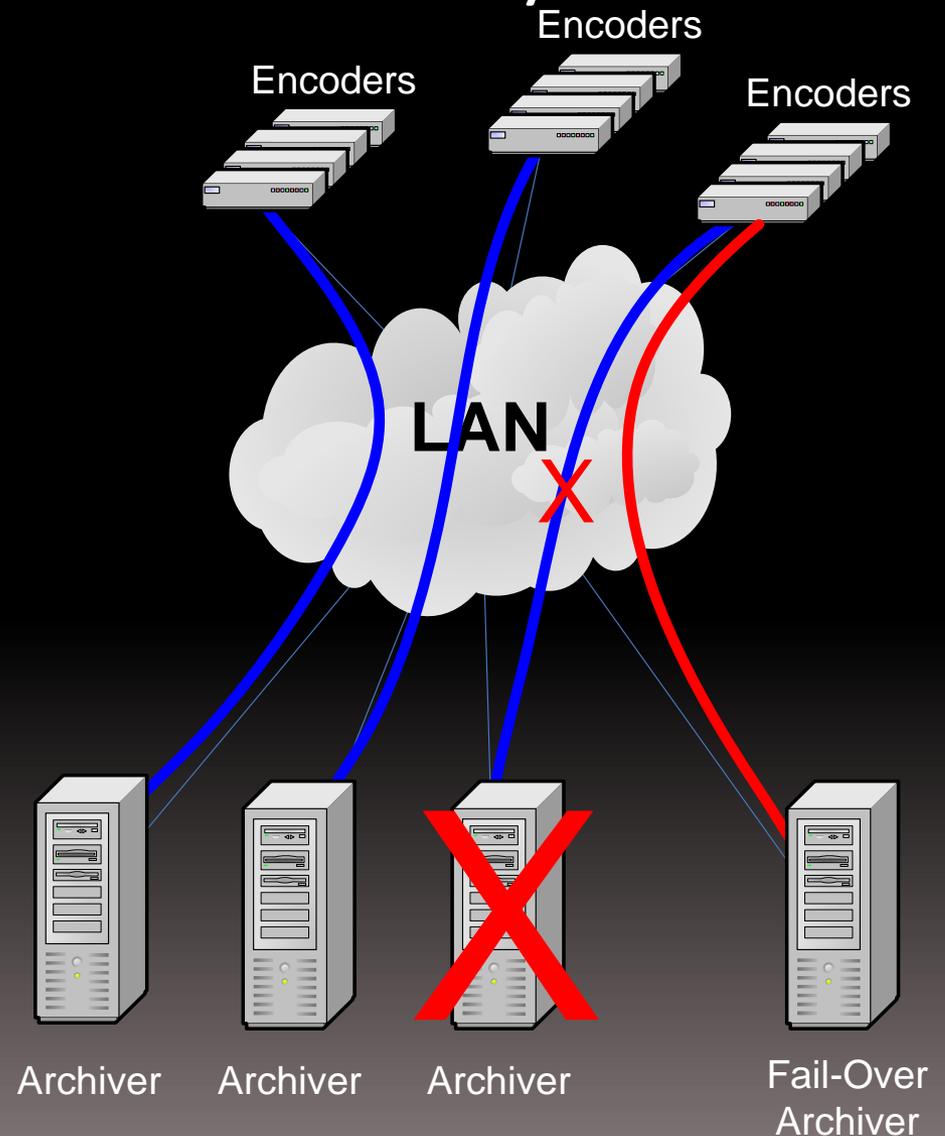
# Key Feature: Redundancy



**Failover Directory**

**Failover/Redundant Archiver**

**Failover Virtual Matrix**



# Key Feature: Software Dev Kit

ソフトウェア開発キット

- » パートナーがスタンドアロンアプリケーションを作成可能
- » ライブ/プレイバックビデオ
- » イベント取り扱い
- » アラームコントロール
- » ユーザーインターフェイスコンポーネンツ

# Key Feature: IT Friendly

## 容易な実装

- » **Active Directory 統合**
- » **効率的な大量実装にはサイレントインストールをサポート**
- » **IPカメラ/エンコーダ自動検出**
- » **すべてのカメラへテンプレートの設定コピー可能**
- » **直感的で総合的なユーザインターフェイス**

# Metadata

- 解析
- GPS
- 暗号化 (DRM)
- プロセスコントロール
- アクセスコントロール
- インテリジェントセンサー



# Organisation shaping technologies

組織を形作る技術

極小

Atom プロセッサ



極大

IPv6  
Federations



信頼できるコンピューティング

暗号化  
デジタル権利管理



Metadata

インテグレーション  
オープンアーキテクチャ  
解析



# Software is the Glue

ソフトウェアは接着剤

“The function of good software  
is to make the complex  
appear to be simple.”

*Grady Booch*

良いソフトウェアの機能とは  
複雑なものをシンプルに  
見せるものです



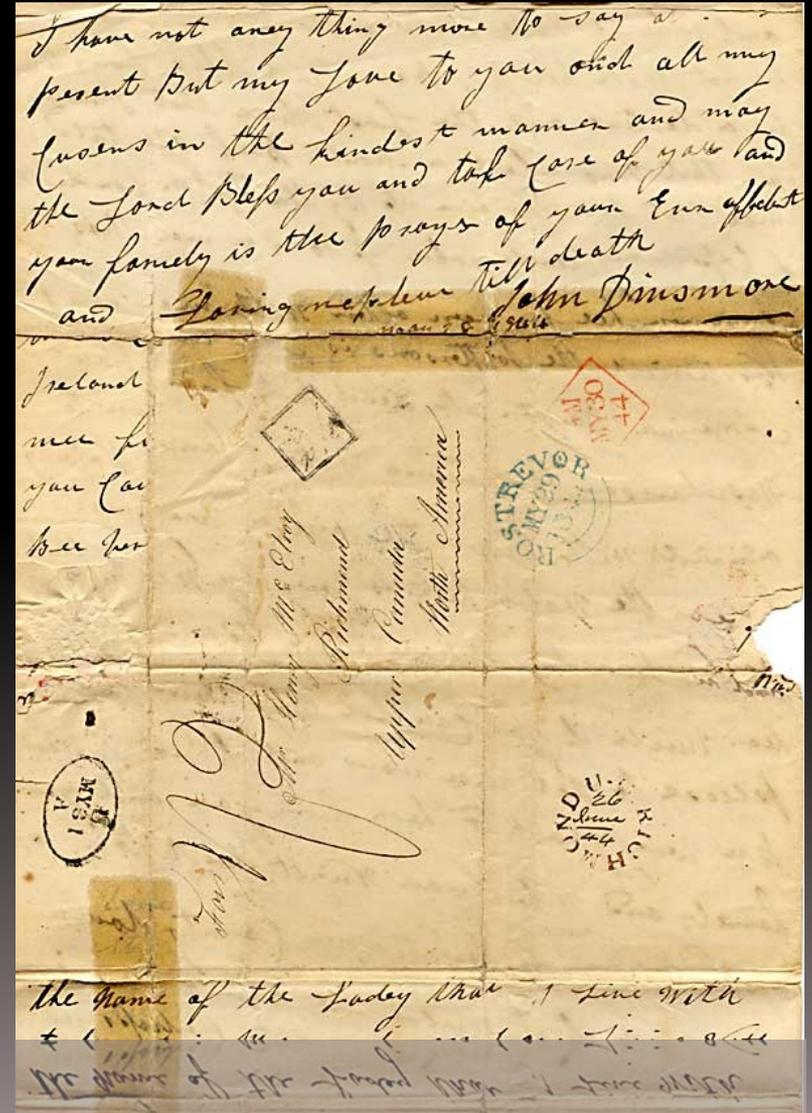
# What is simplicity?

簡潔性とは

“I have made this letter longer because I lack the time to make it shorter.”

*Blaise Pascal, 1657*

長い手紙を書いたのは  
短くする時間がなかったからです



# Télé-Sûreté

- CCTV -> ONV (オープンネットワークビデオ)
- 共有は利益
- コストセンターをプロフィットセンターに
- セキュリティに使われるビデオは20%以下
  - 運営効率最適化
  - マーケティング
  - カスタムリレーション
  - 災害管理

# Enterprise shaping technology

企業を形作る技術

- ソフトウェア
  - ミッションクリティカル
  - すべてのコンポーネントを結び付け
  - すべての、あらゆるハードウェアより長命
  - “ソフトウェアはハードウェアへの投資を守ります”. H Nicolas.
- ワークフロー
  - “プロセスは思考の代わりではなくコミュニケーションの代わりです”
- 統合的な知覚によるインテリジェントな自動化
  - ゴリラがバスケットボールゲーム中を通過した
  - サッカー抑制

# Workflow

ワークフロー

- ルーティング解決
  - ソース
  - 目的地
  - 内容(メタデータ)
  - 危機レベル
  - 瞬間的なロード
  - ユーザ権限
  - トポロジー

# Software Creationism

ソフトウェア作成説

creationism: n. :

革新的なソフトウェアデザインはあらかじめ完全に指定されていて、普通の能力を持ったプログラマーたちのチームの普通の努力で何も無いところから簡単に魔法のように出来上がるという(誤った)信念。

実際、経験から良いデザインは進化と一人の(もしくは一握りの)極めて有能なデザイナーと積極的なユーザー集団の間でなされる検証の相互作用からのみ生まれ—最初の大きな新しいアイデアは常に間違っているということを経験が繰り返し証明している。

—MIT Jargon File

# Last Thought

最後に

“Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.”

*Antoine de Saint-Exupery*

これ以上付け加える物が無くなった時でなく  
これ以上取り去る物が無くなった時が完成だ

Thank you  
ありがとうございました

Questions?