

# SynViz S2 の要員計画機能の開発

## Development of Resource Planning Capabilities for SynViz S2

(株)日立ソリューションズ東日本のプロジェクト管理プラットフォーム製品である SynViz S2 は、バージョン 2.0 で要員計画機能の大幅な強化を行っている。

これまで SynViz S2 が主なターゲットとしてきたプロジェクト管理の業務では、作業（アクティビティ）を主体として、日程計画を立案し、実績が計画どおりに進捗しているかを管理する。一方、要員計画の分野では、人や設備（リソース）を主体として、過不足なく効率的に作業を割り当てていくことが主要な業務となる。

バージョン 2.0 では、SynViz S2 が強みとする高度な表現力と操作性を備えたガントチャート画面を活かして、要員計画分野の業務をサポートする新機能を追加し、適用領域の拡大をめざす。

竹本 宗弘	Takemoto Munehiro
太齋 真吾	Dasai Shingo
太田 卓也	Ota Takuya
石川 裕之	Ishikawa Hiroyuki
門司 太郎	Monji Taro
庄子 真海	Shoji Masami

### 1. はじめに

SynViz S2 は、複数の場所・システムに分散するプロジェクトに関連する情報を集約・統合し、それらをビジュアル化しタイムリーに提供する情報基盤(プラットフォーム)となることをコンセプトとした Web アプリケーションである。

表現力・操作性の高いガントチャート画面を備えており、Web API とカスタム属性を利用することで、他システムとの柔軟な連携を実現できる<sup>1)</sup>。

バージョン 2.0 では、適用領域の拡大に向け、要員計画機能の大幅な強化を行っている。要員計画では、プロジェクトではなく人や設備（リソース）が管理の主軸となり、主に以下のような業務がターゲットとなる。

#### (1) 製造業の設計業務

設計担当者（リソース）は、同時期に複数の製品の設計を受け持っており、設計部門の管理者は、納期や優先度を考慮し、複数のプロジェクトをまたいで設計担当者の割り当てを決定する必要がある。

#### (2) 個別受注生産

工作機械、作業場所など限られた生産設備や作業員（リ

ソース）を複数の案件で共有しており、1 日より短い時間単位で割り当てを調整する必要がある。

#### (3) 設備工事・保守

トラブル対応など計画外の突発的な作業が発生する場合があります。対応可能な作業員（リソース）を素早く探し出し割り当てる必要がある。

#### (4) プラントなどの設備保全

数百/週の作業を、法令などにしたがって適切な資格を持った作業員（リソース）に割り当てる必要がある。

本稿では、SynViz S2 の要員計画機能の紹介と、製品化における課題と対策について述べる。

## 2. 要員計画機能

### 2.1 プロジェクト指向とリソース指向

要員計画では、現行の SynViz S2 が主要ターゲットとしているプロジェクト管理の分野とは、管理と操作の視点が異なる。表 1 に、両者の違いをプロジェクト指向とリソース指向という観点で整理した結果を示す。

表 1 プロジェクト指向とリソース指向

	プロジェクト指向	リソース指向
計画の管理者	プロジェクトマネージャー	部門・拠点の管理者
主な用途	作業計画の作成と進捗管理	要員割当と負荷調整
日程の時間粒度	大・中日程(日・週・月)	小日程(シフト・時間単位)
初期データ作成方法	標準 WBS から作成, 過去の類似プロジェクトから作成	定期作業投入, 上位システムから自動取得

プロジェクト指向では、作業計画はプロジェクト単位で管理され、プロジェクトの大・中日程の計画立案そのものが管理者の主な業務となる。

一方、リソース指向では、立案済みの大・中日程を入力し、部門や拠点の視点で作業を担当者や設備へ割り当てることが主な業務となる。

例えば、プラントなどの設備保全の計画では、上位システムで立案済みの作業計画をもとに、拠点の管理者が拠点内の要員の資格や負荷を見ながら作業の割り当てを行う。

製造業の設計業務では、設計を専任する部門の管理者が、製品ごとに立案済みの大・中日程から、設計工程の作業のみを対象として部門内の設計担当者に割り当てる。設計担当者は同時期に複数の製品の設計を受け持つため、計画が変更になった場合はプロジェクトをまたいで日程や担当者を調整する必要がある。

また、個別受注生産や設備工事・保守の計画では、シフトや時間単位など、日より細かい粒度で作業計画を管理する必要がある。

これらの結果から、特に重要な要件として以下の3点を抽出した。

- (1) リソース軸での作業割当状況の可視化と、簡単な操作による調整
- (2) 複数のプロジェクトをまたいだ日程や作業割当の調整
- (3) 日より細かい粒度での作業計画の作成

次節では、これらの要件をもとに設計した要員計画の主要な3つの機能を紹介する。

## 2.2 リソース別ガントチャート

現行の SynViz S2 のガントチャート画面は、上段にタスク別ガントチャート、下段にリソースグラフを配した二段構成となっている。

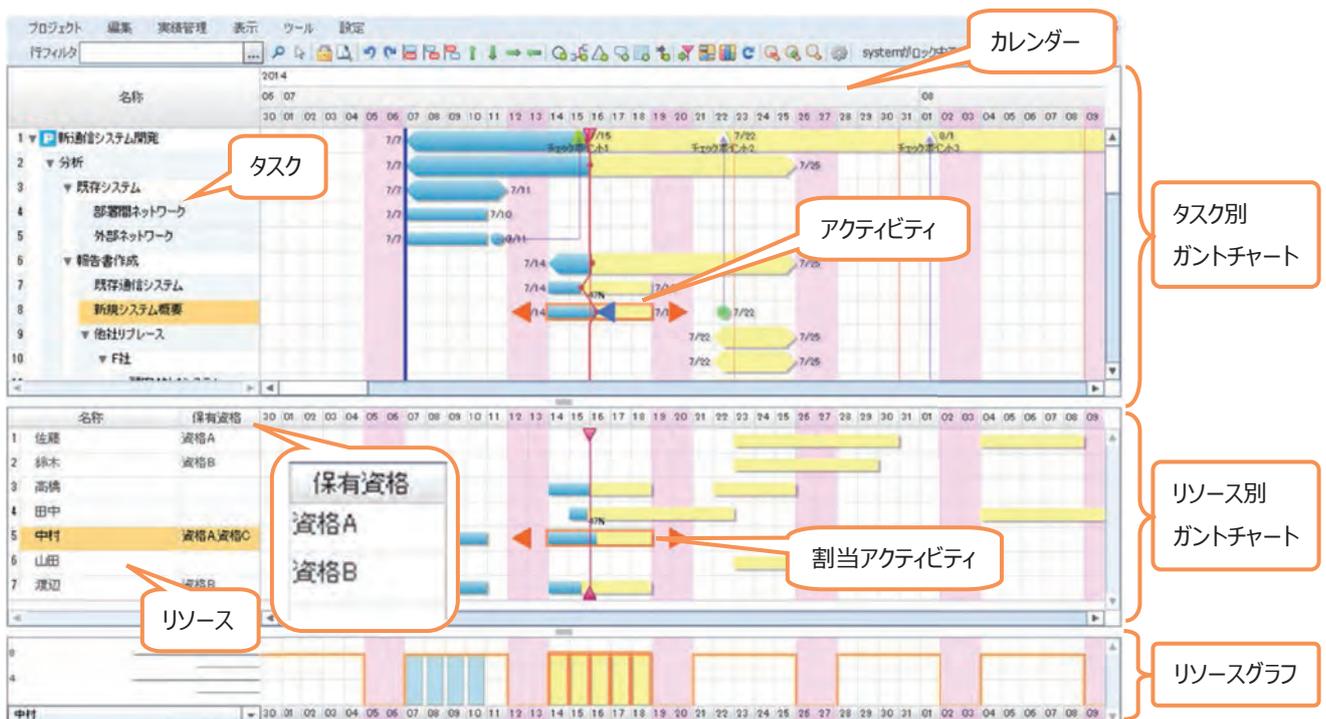


図 1 SynViz S2 v2.0 のガントチャート画面

タスク別ガントチャートの縦軸は「タスク」と呼ばれる階層化された作業項目の一覧であり、横軸はカレンダーである。それらに囲まれた中央の領域に、各タスクの作業期間を示す「アクティビティ」と呼ぶバーが表示される。また、リソースグラフには、単一のリソースの作業の負荷の大きさが表示される。

現行の画面でも、単一のリソースの負荷状況を見ながら日程の調整を行うことはできたが、複数のリソースの作業割り当て状況を俯瞰することができなかった。

バージョン 2.0 では、中段にリソース別ガントチャートを追加した(図 1)。リソース別ガントチャートの縦軸はプロジェクトの全リソースの一覧であり、横軸は各リソースのカレンダーである。それらに囲まれた中央の領域に、各リソースに割り当てた作業期間を示す「割当アクティビティ」と呼ぶバーが表示される。

また、リソース別ガントチャートを別の画面とするのではなく、同じ画面の中にタスク別ガントチャートと並べて表示し、カレンダーの横スクロールが連動するようにしたことが大きな特徴となっている。上段のアクティビティを選択すると中段の対応する割当アクティビティが、中段の割当アクティビティを選択すると上段の対応するアクティビティが自動的に選択状態となるため、タスクとリソースの対応が見つけやすく、マウスやキーボードを使用した直感的な操作で未割当作業のリソースへ

の割り当てや、作業の割当先リソースの変更を行うことができる。

リソース別ガントチャートの縦軸には、追加列としてリソースのカスタム属性<sup>1</sup>を表示することができるため、例えば「保有資格」のカスタム属性を追加することで、要員の資格を見ながら作業の割り当てを行うことができる。

### 2.3 アクティビティチャート

アクティビティチャートは、複数のプロジェクトから特定の条件で抽出したアクティビティ(作業)に対し、リソースへの作業の割り当て、リソース間の負荷の調整、および日程の微調整を行うための画面である。

図 2 にアクティビティチャートの表示例を示す。上段のガントチャートには複数のプロジェクトから抽出したアクティビティを属性ごとに集約して表示する。

図の例では、アクティビティにカスタム属性「工程」「優先度」「部品コード」を追加し、工程=設計のアクティビティを、優先度と部品コードで集約している。

中段のリソース別ガントチャートの縦軸には、任意に選択したリソースの一覧を表示することができる。

業種にあわせてカスタム属性を追加することで、場所や必要な資格など、様々な条件でアクティビティを抽出・集約し、プロジェクトをまたいで日程や負荷の調整

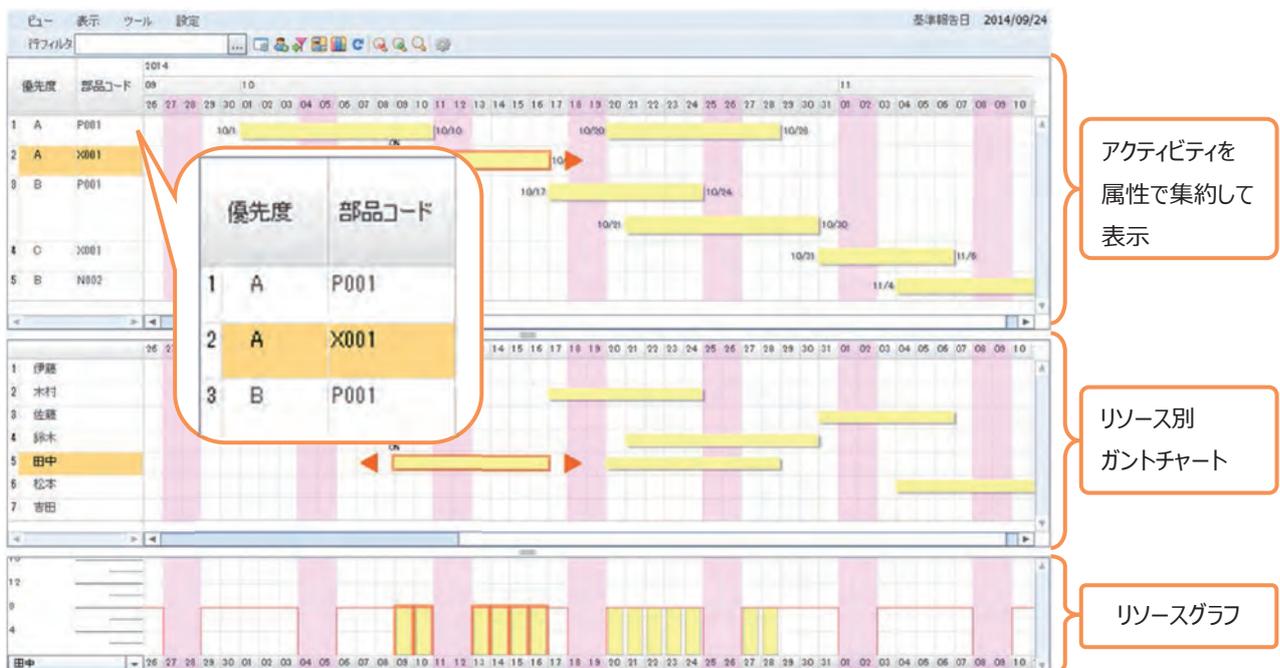


図 2 アクティビティチャート画面

<sup>1</sup> ユーザーが任意に追加可能な付帯情報

を行うことができる。

また、アクティビティのコンテキストメニューから、当該アクティビティの属するプロジェクトのガントチャート画面に遷移することができるため、大・中日程の全体の状況や、他部門が担当する先行作業の進捗状況を確認することもできる。

## 2.4 タイムスライス

タイムスライスは、1 日を複数のバケットに分割し、日よりも細かい粒度で日程計画を作成できる機能である。

日よりも細かい粒度への対応方式には、「時分単位」と「時間枠 (タイムスライス)」の 2 つの方式があった。

時分単位の方式は、単純にガントチャートの横軸の単位を日単位から時刻単位にする方式である。この方式には、直感的にわかりやすく、どのような時間粒度にも対応できるというメリットがある反面、画面上で利用されない領域が多くなり、視認性・操作性が落ちるというデメリットがあった。例えば、1 日を 8:00 から 12:00 と 13:00 から 17:00 で 2 分割し、午前・午後の単位で計画したい場合、17:00 から 24:00 までの、全画面の 6 割以上が使用されない領域になってしまう (図 3)。



図 3 時分単位方式の表示例

タイムスライス方式では、1 日を N 個の時間枠 (シフト) に分割して計画を表現する。例えば、午前・午後で計画を立てる場合、N=2 になる。シフトは必ずしも 1 日 24 時間を N 個に等分割したものではなく、午前=8:00~12:00、午後=13:00~17:00 のように対応する時間を指定できる。ただし、作業をガントチャート上のバーとして表現するときはこれらの実時刻を用いず、各シフトの長さを均等として 1 日を N 個に等分割した形で表示する (図 4)。

現実には、時分単位で計画を立てることは少なく、午前・午後やシフトのように一定の時間枠を設定し、この枠に対して作業を当てはめていくことの方が多いと考え

られるため、タイムスライス方式を採用した。

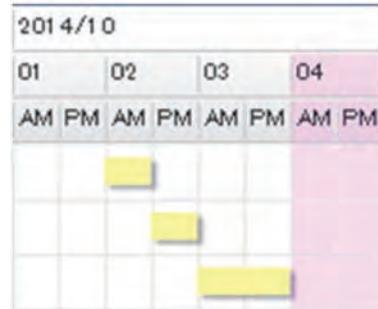


図 4 タイムスライス方式の表示例

なお、N は顧客ごとに業態にあわせて別々の値を設定できるため、N=24 とすることで、横軸を実時刻とする使い方にも対応することができる。

## 3. 製品化における課題と対策

### 3.1 互換性の確保

SynViz S2 は日程計画の参照と編集に必要なすべての機能を REST (Representational State Transfer)<sup>2)</sup>形式の Web API として公開している。このため、画面に対する操作は、基本的にすべて WebAPI を利用してプログラムから実行することができる。

Web API は、外部システムとの連携などの用途で、すでに多くの顧客で使用されており、現行バージョンとの後方互換性を維持したまま、要員計画機能を実現する必要があった。

#### (1) プロジェクト横断 API

現行の Web API はプロジェクト指向で設計されており、一部の例外的なメソッドを除き、計画データはプロジェクト単位でしかアクセスできない。例えば、アクティビティのリストを取得する Web API のリクエストの URI は以下ようになる。

```
GET http://s2app/api/projects/p001/activities?filter=["Name.eq.A"]
```

この例では、プロジェクト ID (プロジェクトを特定するキー) が p001 であるプロジェクトから抽出した名称が A のアクティビティのリストが取得される。

このように、アクティビティのリストを取得する Web API では、プロジェクト ID の指定が必須となっていた。

リソース指向では、複数プロジェクト横断でアクティ

ビティのリストを取得する必要があるが、現行の Web API では、1 回のリクエストで 1 プロジェクトずつしかデータを取得できない。

この課題を解決するために、GET メソッドではプロジェクト ID に“\_\_all”という予約語を指定可能とし、“\_\_all”が指定されたときは全プロジェクトからデータを取得できるように設計した。

SynViz S2 では“\_”で始まる文字列はシステムの予約語を表し、プロジェクトなどの ID には使用できない仕様となっているため、“\_\_all”が既存のプロジェクトの ID と重複することはない。API の呼び出し例を以下に示す。

```
GET http://s2app/api/projects/__all/activities?filter=["Name.eq.A"]
```

この例では、実行ユーザが参照可能なすべてのプロジェクトから抽出した名称が A であるアクティビティのリストが取得される。

また、“\_\_all”のときのみ有効なオプション“projects”を設けることで、特定の複数プロジェクトを対象とすることもできるようにした。

以下の例では、プロジェクト ID が p001 または p002 のプロジェクトから抽出した名称が A であるアクティビティのリストが取得される。

```
GET http://s2app/api/projects/__all/activities?filter=["Name.eq.A"]&projects=p001,p002
```

このように、プロジェクト ID の部分に特殊な文字列“\_\_all”を指定する方式を採用することで、後方互換性を維持しながら、プロジェクト横断でデータを取得する API を追加することができた。また、実装の点でも、既存 API への変更が最小限に抑えられたため、既存 API 部分の性能と品質の確保が容易であった。

## (2) タイムスライス

個別受注生産では、工程ごとに管理する時間単位が異なり、特定の工程のみタイムスライスを使用するケースがあった。例えば、設計工程は日単位で、検査工程はシフト単位で管理する。

そのため、アクティビティ単位でタイムスライスを指

定可能とした。WebAPI では、アクティビティオブジェクトにオプションな属性としてタイムスライスの情報を追加し、省略時は日単位のアクティビティとして扱うようにすることで、後方互換性を維持した。

## 3.2 性能の確保

### (1) ガントチャート画面

2.2 で示したとおり、バージョン 2.0 では、ガントチャート画面にリソース別ガントチャートを追加した。

リソース別ガントチャートでは、プロジェクトの全リソースに対して、他プロジェクトの作業割当の情報も含めて表示する。これらを取得・表示しながら、前バージョンと同等の初期表示性能を維持することは難しい。

そこで、図 5 に示すように、初期表示時にリソース別ガントチャートを非表示としている場合はリソース別ガントチャートの表示情報をロードせず、中段を表示したときに初めて情報を取得するようにした。

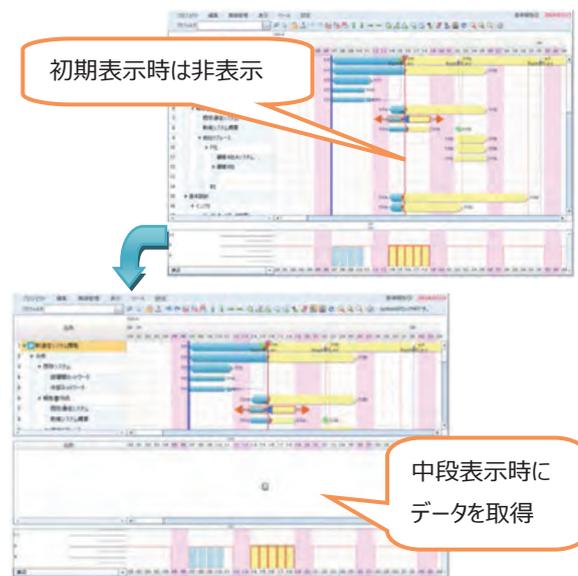


図 5 ガントチャートの初期表示性能確保の施策

中段を除くと、初期表示時に取得・表示するデータの量は、機能エンハンスにより追加されたいくつかの属性は増えているもののその差はわずかであり、前バージョンとほぼ同等の性能を確保できた。

また、ガントチャート画面では前回の表示状態を記憶し次回表示時に復元する仕様であるが、図 6 のオプション設定により、性能を優先したい場合はリソース別ガントチャートを前回の表示状態に関わらず、初期表示時は常に非表示にできるようにした。

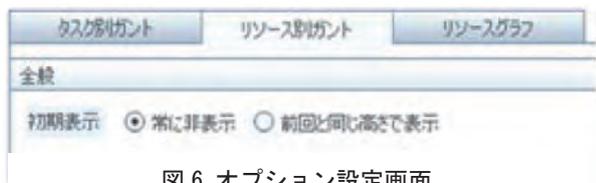


図 6 オプション設定画面

れまでターゲットとしてきたプロジェクト指向の業種に加え、リソース指向の業種にも適用することが可能となった。

今後は、プロジェクト管理の領域に加えて、製造業の設計業務、個別受注生産、設備工事・保守、保全などの領域へ適用を拡大していく。

## (2) アクティビティチャート画面

アクティビティチャート画面では、複数のプロジェクトからアクティビティを抽出し、表示する必要がある。1つのプロジェクトを対象とするガントチャート画面よりも検索対象の母数が大きくなるため、無条件に同等の性能を確保することは難しい。

そこで、3.1(1)に示したプロジェクト横断 API に属性と期間による絞り込み機能を充実させ、エンドユーザ側で必要十分なデータのみ取得し、応答性・操作性の劣化を回避できるようにした。図 7 にアクティビティチャート画面の検索条件指定画面を示す。

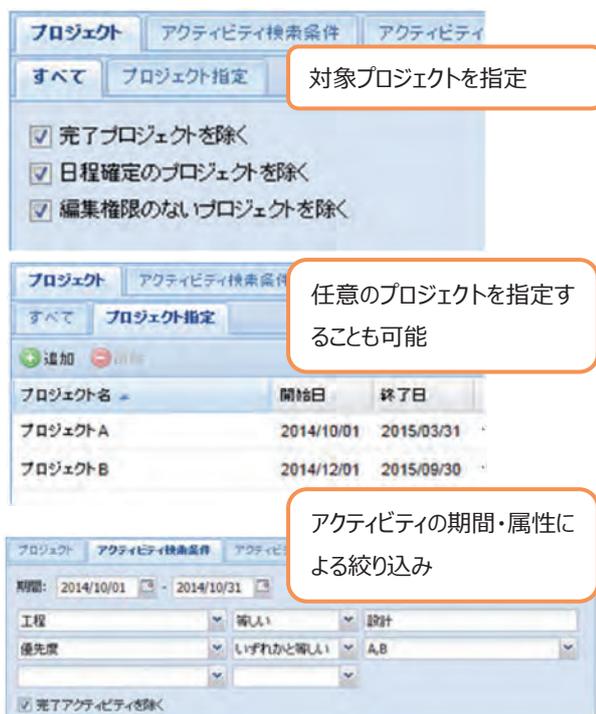


図 7 アクティビティチャートの検索条件指定画面

## 4. おわりに

SynViz S2 はバージョン 2.0 において、要員計画機能を大幅に強化した。

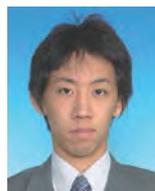
リソース別ガントチャート, アクティビティチャート, およびタイムスライスの機能を追加したことにより、こ

参考文献

- 1) 庄子真海 他: プロジェクト情報基盤としての SynViz S2 による複合ソリューションの実現, 日立ソリューションズ東日本技報 第 19 号, 2013
- 2) R. Fielding: Architectural Styles and the Design of Network-based Software Architectures, Chapter 5 Representational State Transfer, 2000, [http://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm), (2014 年 9 月閲覧)



竹本 宗弘 2000 年入社  
パッケージ開発部  
S2 設計開発  
munehiro.takemoto.kk@hitachi-solutions.com



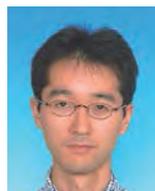
太齋 真吾 2002 年入社  
パッケージ開発部  
Flex 版ガントチャートコンポーネント設計開発  
shingo.dasai.dc@hitachi-solutions.com



太田 卓也 2010 年入社  
パッケージ開発部  
S2 設計開発  
takuya.ota.fd@hitachi-solutions.com



石川 裕之 2000 年入社  
Viz ソリューション部  
S2 ソリューション適用  
hiroyuki.ishikawa.ha@hitachi-solutions.com



門司 太郎 1994 年入社  
パッケージ開発部  
S2 アーキテクチャ設計  
taro.monji.zc@hitachi-solutions.com



庄子 真海 1999 年入社  
パッケージ開発部  
SynViz S2 プロダクトマネージャ  
masami.shoji.wg@hitachi-solutions.com