

# Where Am I?

SLAM アルゴリズムを利用  
した環境認識

自動運転の構想を現実のものとするためには、車両の位置をあらゆる環境のあらゆる時点において常に認識できるようにする必要があります。また、これは詳細な地図や衛星ナビゲーションが利用できない場合でも必要です。このような場合、SLAM アルゴリズムは優れたソリューションを提供します。

**自** 動運転にとって、周辺環境における車両の正確な位置を認識することは重要な前提条件です。最も一般的なソリューションは、正確な地図と衛星または慣性ナビゲーションシステム（慣性計測ユニット、IMU）、および推測航法（移動方向と速度を利用した位置認識）に

よるナビゲーションを使用することです。ただし、現状のこれらのシステムでは詳細な地図をつなぎ合わせたネットワークしか利用できないうえに、衛星支援測位システムの精度も不十分な場合が多々あります。この問題の解決策となるのは、各種の SLAM (Simultaneous Localization

And Mapping) アルゴリズムです。これらのアルゴリズムは、LiDAR などの環境センサから受信したデータを利用して車両環境全体の 3D マップを生成し、周辺環境における車両の位置を特定します。この場合、取得した 3D データと比較したり調整したりできるおおまかな地図データな



ど、現在の環境で入手できるデータが多ければ、それだけアルゴリズムの動作精度が向上します。

### RTMaps における SLAM アルゴリズムの使用

マルチセンサ開発環境である RTMaps (Real-Time Multisensor applications、情報ボックス「製品の特長：Intempora 社の RTMaps」を参照) では、あらかじめ設定されたコンポーネントライブラリが含まれており、Dibotics 社の Augmented LiDAR 3D SLAM アルゴリズムを利用できます。RTMaps は、dSPACE ツールチェーンに統合されています。SLAM アルゴリズムは、センサデータベースの認知および位置特定アルゴリズムであり、センサフュージョンもオドメトリ (ホイール回転など、推進システムを利用した位置推定) も必要ありません。SLAM アルゴリズムでは、わずか 1 つの LiDAR センサのデータを基に、多数のデータセットを組み合わせることで 3D モデルをリアルタイムに生成します。3D モデルはさまざまな視点から確認することができます。また、検出したオブジェクトの分類情報も含まれており (図 1)、周辺環境における自車の位置の特定も可能です。さらに、環境モデルはその後の開発ステップの土台として使用することができ、軌道計画やモーションコントロールなどの自動運转向けアプリケーションを開発するうえでも有用です。RTMaps の SLAM ライブラリは、テストドライブ中に取得したセンサデータを再生する際に使用します。RTMaps は、単純かつ直感的に使用できるブロックベースの開発環境です。そのため、最初にコンポーネントライブラリから必要なコンポーネントをドラッグアンドドロップでダイアグラムに追加し、次にコンポーネントをダイアログに従ってパラメータ化し、最終ステップで必要に応じてデータリンク経由で接続を行います。あとはボタンをクリックするだけでアプリケーションを実行およびテストできるようになります。ここでは、記憶媒体に保存した生データが再生ブロックを

介して再生され、SLAM アルゴリズムおよびセグメント分割アルゴリズムに転送されます。アルゴリズムの出力は、3D ビジュアル表示ブロックによってグラフィカルに表示されます。マウスを使用して操作すれば、3D モデルの視点を変更することもできます。RTMaps にはマルチスレッド機能と効率的なメモリ管理機能が搭載されているため、複雑なアルゴリズムであっても非常に効率良く実行することができます。

### 車両での自動運転機能の開発

車両でアルゴリズムのプロトタイピングとテストを迅速に行うには、アプリケーションを実際の環境センサに接続する必要がありますが、これは、RTMaps ダイアグラムの再生ブロックを、センサデータを取得するコンポーネントに置き換えるだけで実施できます。この開発環境にはカメラ、レーダー、LiDAR、および車両バスを接続するためのさまざまなインターフェースが用意されており、タイムスタンプを発行して時間相関性のあるセンサデータを取得、処理、および再生することも可能です。自動運転機能の開発では、生成した 3D 環境モデルの情報を処理するために状況分析や軌道計画などの特定のアルゴリズムを頻繁に使用します。RTMaps では、これらのアルゴリズムを C++、Python、または Simulink® コード形式の独自コンポーネントとして統合することができます。開発者は、このような独自のアルゴリズムを組み込む際に、Windows® および Linux の SDK ウィザードやサポート用の多数のコード例を使用できます。車載開発プラットフォームである dSPACE MicroAutoBox 上で実行される制御アルゴリズムとの間で必要なデータのやり取りは、モーションコントロールなどを通じて直接行うことができます。dSPACE ツールチェーンに RTMaps をシームレスに統合するには特別なコンポーネントライブラリを使用します。これにより、データの時間同期転送を dSPACE リアルタイムプラットフォームで行えるようになります。 >>

RTMaps は、自動運转向けのアルゴリズムの作成に必要なすべてのステップを備え、組込みプラットフォーム上での開発作業を実現することにより、開発者をサポートします。

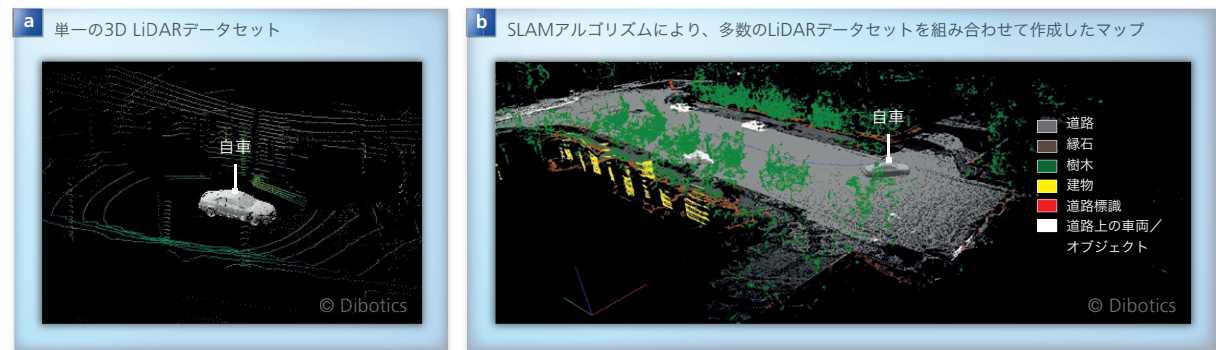


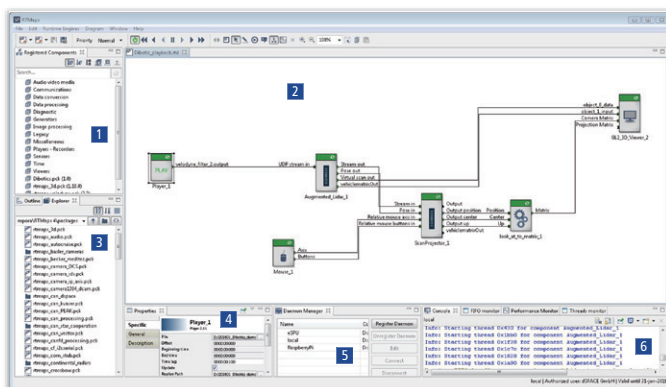
図1：個々の未処理のLiDARポイントクラウド (a) および多数のLiDAR データセットから構築した3Dマップ (b) の比較により、SLAM アルゴリズムの有効性が明らかに示されています。

### PC から組み込みプラットフォームまで継続的に作業

開発プラットフォーム上に演算処理リソースやハードウェアアクセラレーション用のリソースなどが存在すると、複雑なアルゴリズムを実行する際の動作に影響を与える可能性があります。このため、開発者の多くは、可能な限り早期にターゲットプラットフォーム上で開発を行いたいと考えています。このような場合に RTMaps の便利なツールを利用すれば、開発者は選択するプラットフォームにかかわらず、常に PC 上の使い慣れた環境で開発を行えます。Remote Studio Connector (図3) を使用すると、キーボード、マウス、またはモニタを接続することなく、プラットフォームから TCP/IP 接続経由で直接 RTMaps ダイアグラムを処理および実行することが

できます。また、開発者は常に選択したシステムに適したコンポーネントライブラリを使用して作業することができ、選択したプラットフォームのデータシステムから直接アプリケーションをロードしたり保存したりできるため、作業環境の利便性も自動的に向上します。さらに、パラメータ設定オプションを活用すれば、ターゲットプラットフォームをオンに切り替えると同時にランタイム環境で目的のアプリケーションをロードし作業を開始することも可能です。RTMaps Studio に組み込みプラットフォームのサポートを追加すると、開発者は使い慣れた自分の PC 環境から組み込みプラットフォームを簡単かつ便利に操作できるように、自動運転機能のプロトタイピングをこれまでになく高品質で実行することができます。■

図2：RTMaps のユーザーインターフェース。モジュール型の開発環境により、幅広いマルチセンサアプリケーションを容易に処理することができます。そのために、総合的なコンポーネントライブラリが用意されています。



- 1 **コンポーネントライブラリ** – RTMaps 環境で登録されている、または利用可能なコンポーネントモデルを表示します。これらのコンポーネントモデルは、(センサ、ビューフ、画像処理などの)機能ごとに分類されます。
- 2 **RTMaps ダイアグラム** – アプリケーションダイアグラムの作成に使用されます。この例で使用されているコンポーネントは、LiDAR データの再生用ブロック、Dibotics 社製の SLAM およびセグメンテーション用コンポーネント、視点を設定するためのマウス制御、および 3D ビジュアル表示です。
- 3 **Explorer および Outline** – Explorer：コンピュータ上のローカルディレクトリのビューを提供します。通常は、RTMaps に登録できるパッケージが入ったディレクトリが表示されます。Outline：このビューには、一瞥とクイックナビゲーション用のダイアグラムスペース全体が表示されます。
- 4 **Properties** – 現在のダイアグラムにさまざまなパラメータを設定することができます。
- 5 **Daemon Manager** – 組み込みプラットフォームに接続し、リモートダイアグラムを開いて処理および実行することができます。
- 6 **Console** – コマンド、コンポーネントメッセージ、警告、エラーなど、さまざまなテーマに関する情報を提供します。

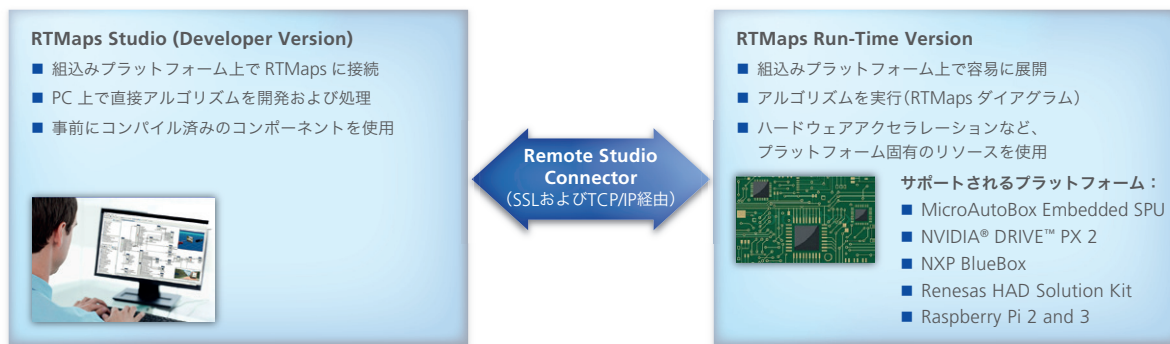
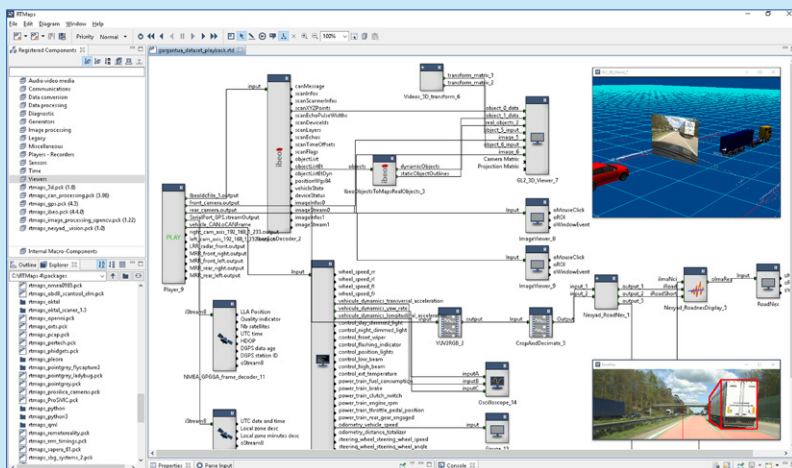


図 3 : RTMaps に Remote Studio Connector を搭載することにより、開発者は通常の PC 上の作業環境を維持したまま、さまざまな組み込みプラットフォーム上で組み込みプロトタイピングを実行することができます。

## 製品の特長 : Intempora 社の RTMaps



プラットフォームである VEOS 用に特別に設計されたインターフェースブロックセットを提供しています。また、RTMaps を ControlDesk に接続すると、開発したアプリケーションのパラメータ化なども容易に行えます。

### NEW : 組み込みプラットフォーム上で直接プロトタイピング

RTMaps に Remote Studio Connector を搭載すると、組み込みプラットフォーム上で高度な自動運転用アプリケーションのプロトタイピングをさらに迅速に行うことができます。これにより、組み込みプラットフォームにマウスやキーボード、モニタを接続することなく、使い慣れたリモート PC の RTMaps 開発環境から直接ターゲットプラットフォーム上の認知および融合アルゴリズムを容易に開発することができます。

### マルチセンサアプリケーション向けの強力な開発環境

Intempora 社の RTMaps は、複雑なマルチセンサアプリケーションに対応するモジュール型のソフトウェア開発およびランタイム環境であり、dSPACE では 2016 年から配布してきました。RTMaps は自動運転、ロボット産業、航空宇宙産業などの分野で重要な役割を果たしています。開発者は、総合的なコンポーネントライブラリを活用することで、カメラ、レーダー、LiDAR といったさまざまなセンサからのデータを正確に取得、同期、および処理することができます。RTMaps では、ユーザー固有のコンポーネントを C++、Python、また

は Simulink コードとして開発し、容易に統合することができます。また、OpenCV や Dibotics 社の Augmented LiDAR 3D SLAM などの複雑なアルゴリズムにも対応しており、NVIDIA DriveWorks もサポートしています。RTMaps は、独自のマルチスレッド機能と効率性に優れたメモリ管理機能を PC ベースおよび ARM ベースのプラットフォーム上で提供します。

### dSPACE ツールチェーンへのシームレスな統合

dSPACE では、RTMaps を dSPACE ツールチェーンに統合するため、リアルタイムシステムや PC ベースのシミュレーション