

Oldenburg 23.02.2007
Tim English



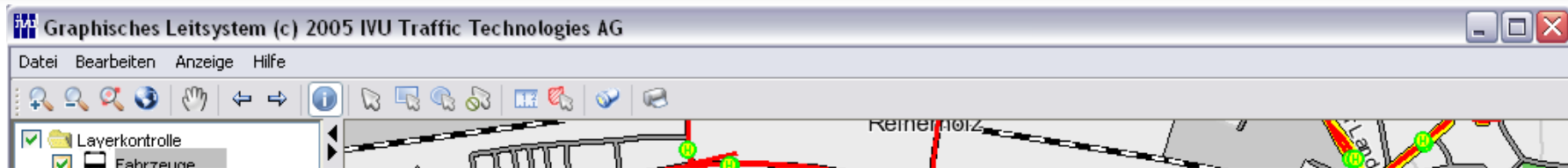
**Dynamische Objekt-Visualisierung in Echtzeit für
Verkehrsleitstellen
auf Basis von Open Source Komponenten**

Agenda



- Anforderungen an BONgis
- Technische Realisierung auf Basis von GeoTools 2.1.1 (GT) und der Java Topology Suite (JTS)
- Livepräsentation
- Releasewechsel auf GT 2.3.0
- Dokumentation von GT 2
- Fazit

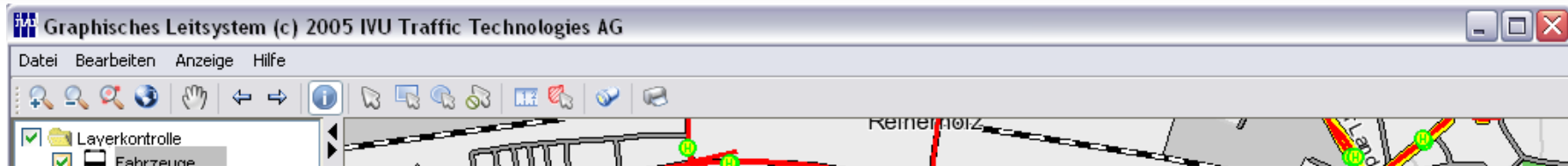
Anforderungen an BONGis



- **Präsentation dynamischer Datenquellen bei der üstra.**
- **Darstellung von mehreren hundert Fahrzeugen mit definiertem Aktualisierungstakt auf einer Basiskarte.**
- **Integration externer Datenquellen (Luftbilder, amtliche Kartenwerke, ...).**
- **Interaktion mit Leitstellensoftware BON**
 - Austausch von Objektstatus (Client)
 - Austausch von Aktionen (Client)
 - Bereitstellung von dynamischen Fahrzeugdaten (Server)
 - Aktualisierung von Liniennetzdaten (Server)



Anforderungen an BONGis



- **Thematische Darstellung der Fahrzeuge entsprechend ihrer Fahrplanabweichung.**
- **Automatische Kartennachführung auf ausgewählten Fahrzeugen (Notfallsituation).**
- **Informationsextraktion über die Karte (Infotool).**
- **Zukunftssicherheit, Nutzung freier Software.**
- **Schaffung und Nutzung offener und fester Schnittstellen durch Einhaltung der OGC-Vorgaben.**

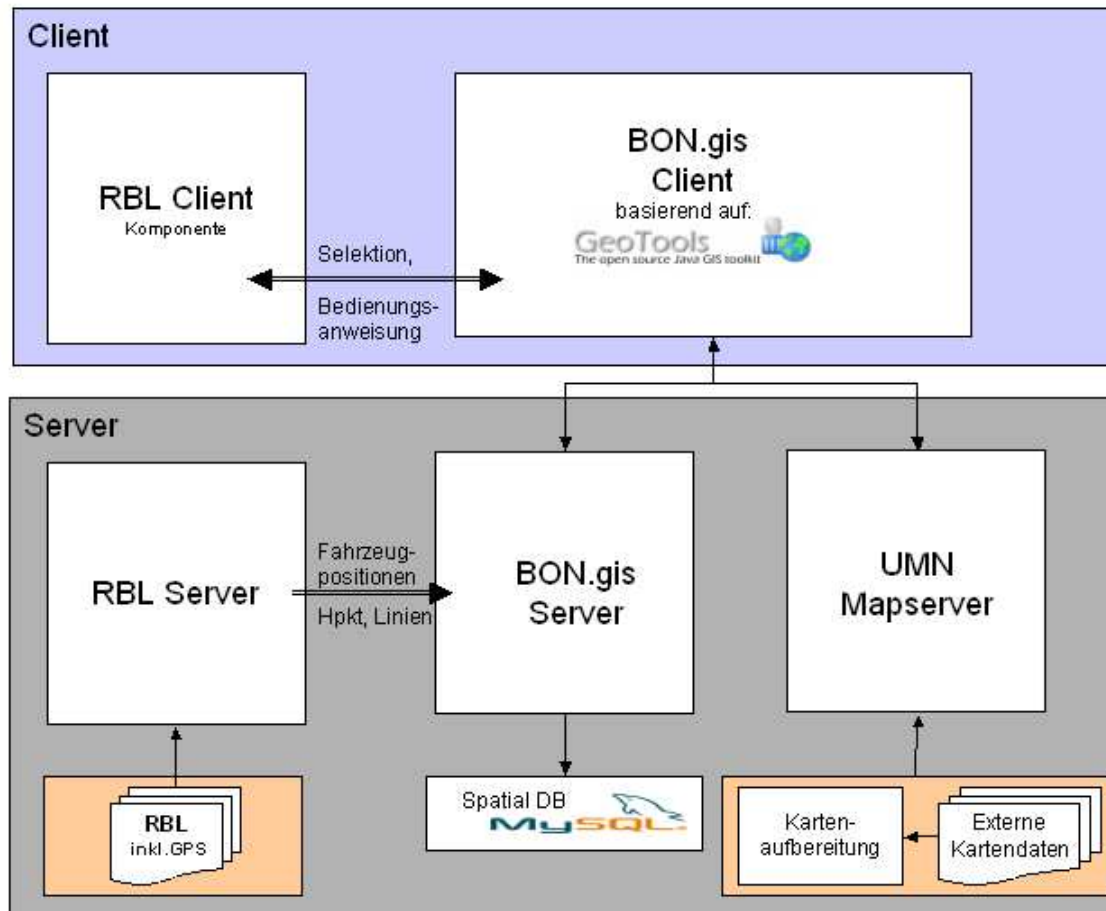




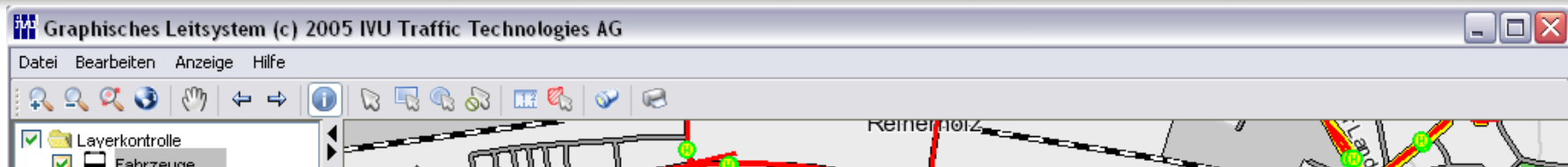
Technische Realisierung auf Basis von GeoTools 2.1.1 und JTS



Systemarchitektur



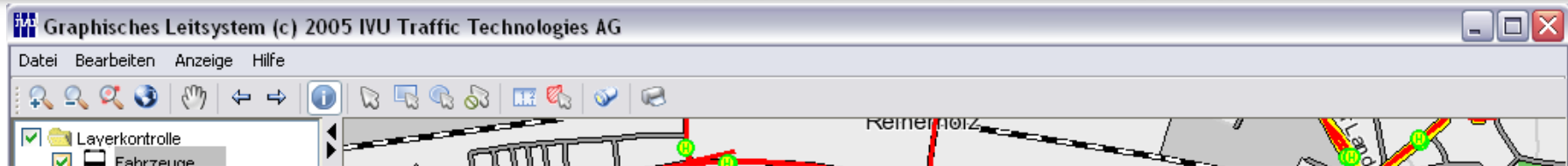
■ Technische Realisierung auf Basis von GeoTools 2.1.1 und JTS



- Die GIS-Lösung ist serverbasiert und läuft in einem Thin-Client.
- Server- und clientseitige Interaktion mit Leitstellensoftware (BON).
- Zentrale Datenhaltung.
- Hybride Architektur
 - Business Objekte als Vektordaten im Client.
 - Beliebige Web Map Services als Hintergrunddienste.
- Unterstützung wichtiger OGC-Standards
 - Web Map Service (WMS)
 - Web Feature Service (WFS)
 - Styled Layer Descriptor (SLD)
 - Simple Feature Implementation Specification (SFS)
 - Filter Encoding Implementation Specification



Technische Realisierung auf Basis von GeoTools 2.1.1 und JTS



■ Einsatz von GeoTools 2 (GT) als GIS-Basiskomponente

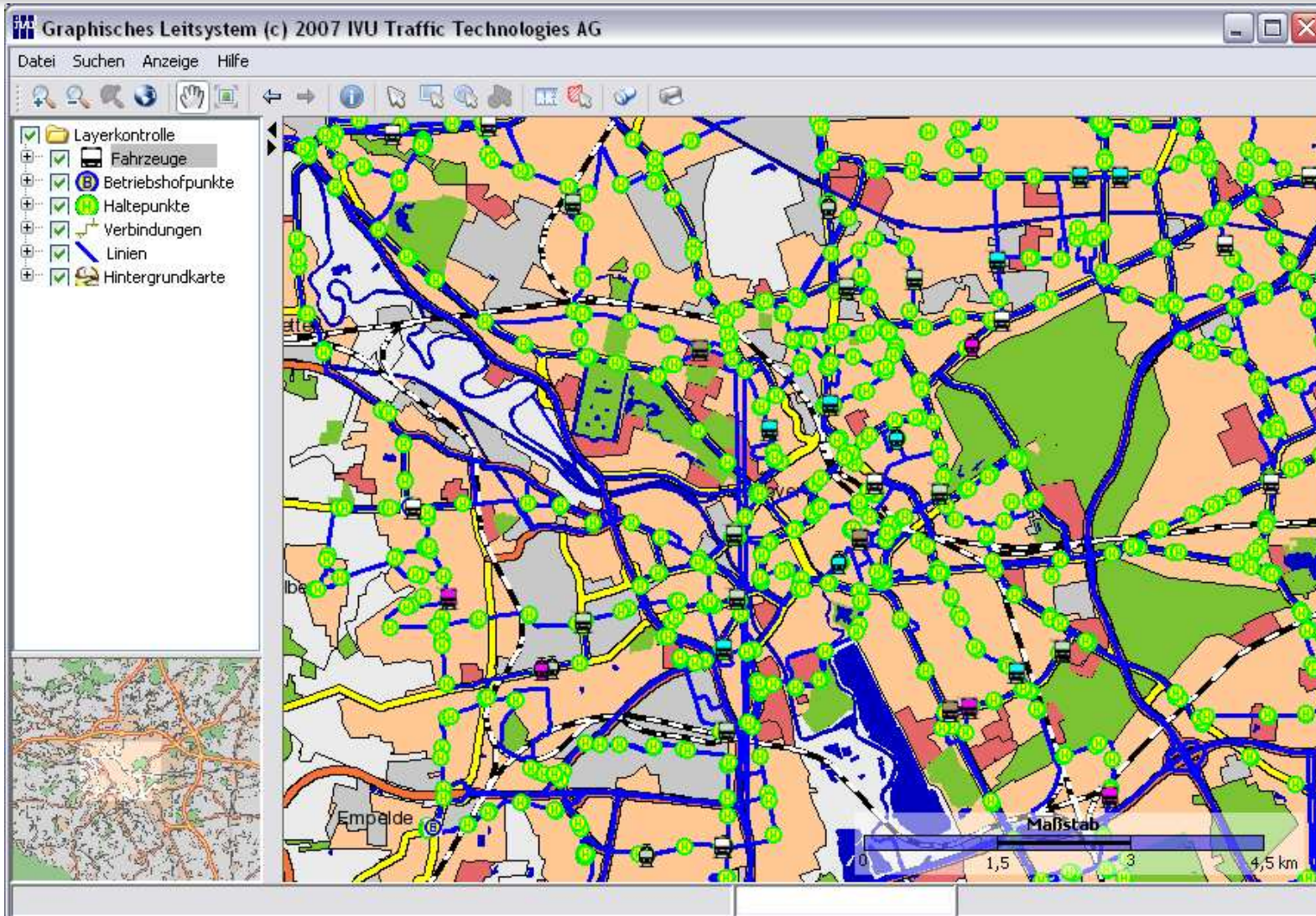
- Nutzung der GUI-Elemente. (StyledMappane, Statusbar, Maßstabsbalken)
- Einsatz der verschiedenen Datastore-Entwicklungen. (WMS, WFS, MySQL, Oracle)
- Erweiterung der Layerschnittstelle.
- Transformation von Geometrien.
- SLD-Parser für das Einlesen der benötigten Stilinformatoren.
- Koordinatensystembereitstellung über EPSG-Codes.

■ Einsatz der Java Topology Suite (JTS)

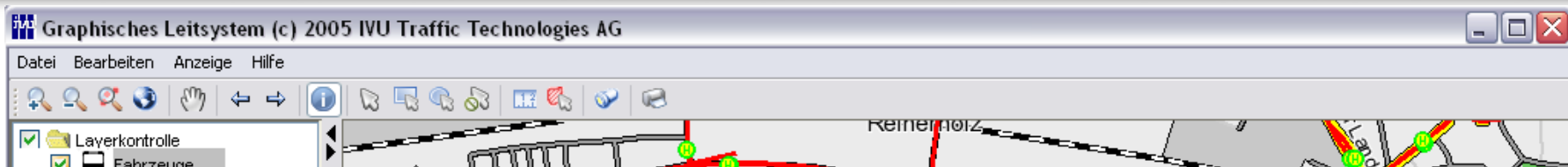
- JTS ist Geometrierepräsentationsbasis von GT2.
- Geometryfactory zum Erzeugen der Fahrzeugobjekte server- und clientseitig.
- Durchführung von Verschneidungen.



Live Präsentation



Releasewechsel von GT 2.1.1 auf GT 2.3.0



■ Verbesserungen

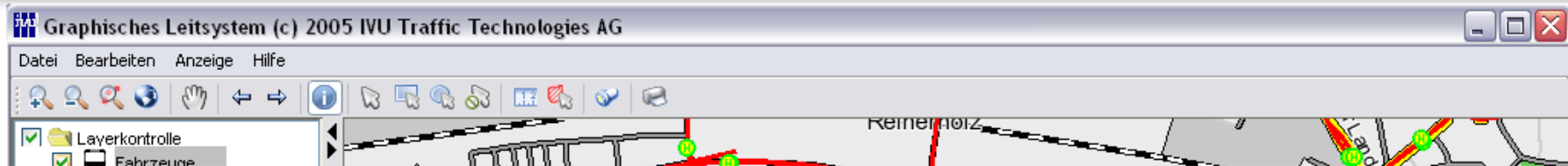
- GT API wurde auf die GeoAPI umgestellt.
- Neustrukturierung der Bibliotheken führt zu besserer Übersicht.
- Performanceverbesserung in der Verarbeitung von Rasterdaten.

■ Probleme

- Sämtliche GUI-Elemente aus GT 2.1.1 waren nicht mehr vorhanden.
- Fehlende Beispiele
- Ausgedünnte Renderer, die nicht so ausgereift „aktualisieren“.
- Filterimplementierungen unterstützen nicht alle Datentypen.
- Fehlende Transparenz in den Änderungen (keine Benennung von aufgelösten Komponenten).

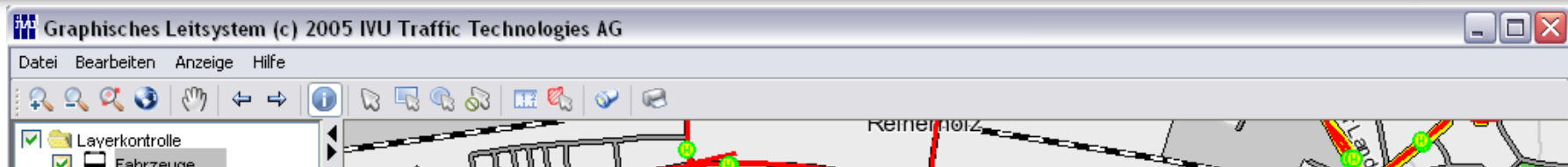


Dokumentation von GeoTools 2



- Ist stark verbesserungswürdig.
- Fehlende Beispiele für neue Versionen.
- Version 2.1.1 und aktuelle Version nicht mehr vergleichbar.
- Aktive Mailinglisten für konkrete Problemlösungen.





- **Sehr gute Basisbibliotheken für individuelle Lösungen.**
- **Der Einstieg ist über die derzeitigen Versionen nur schwer möglich.**
- **Schnittstellenstruktur immer noch stark im Umbruch.**
- **JTS ist eine sehr performante API zur Repräsentation von Geometrien.**
- **Komponenten von GT2 und JTS werden von uns in weiterer Software erfolgreich eingesetzt.**





Vielen Dank für Ihre Aufmerksamkeit!