

cardo4

Für Anwender, Betreuer und Entwickler

René Neidt, IDU IT+Umwelt GmbH

- Erinnerung zu bereits Gesagtem, Resümee
- Migrationsaufwand
- cardo und Iwan
- Anwendungsentwicklung in cardo4
- Galerie

Vortrag von
2015

SOLL

- Berücksichtigung der mannigfaltigen Anforderungen der einzelnen Nutzer
 - ⇒ egal wie viel vorkonfiguriert wird, es wird nie 100%ig für alle passen.
 - ⇒ Sicht auf die Inhalte ist stark vom fachlichen Hintergrund abhängig
- Hochgradig personalisierbar (Stichwort: Projekte)
 - ⇒ Methoden zum „Teilen“ (Share) von Projekten
 - ⇒ Teilen mit ... eigenen Benutzerkreisen

- Alle Vorbereitungen für umfangreiche Möglichkeiten zur Personalisierung / individuellen Konfiguration abgeschlossen
- Alle wesentlichen Features implementiert
- Auf Ihr Veto: Näherung „an klassische Ansicht“ möglich
- „Benutzerkreise“ noch nicht implementiert

SOLL

- Interne Kommunikation der Komponenten über weitere Gegenstände ermöglichen
⇒ bekannte Typen, (Thema für morgen)
- Gleichberechtigte Nutzung von „Geodaten“ und „nicht Geodaten“
- Für konkrete Probleme konkrete Lösungen bieten

„WellknownType“ System erfolgreich abgeschlossen:

IWellKnownType:

- SourceId, AdminTreeld
- IAnyEditableTableData
- IAnyLayer
 - RestorableIwanLayer
- FeatureCollection
- ... beliebige weitere selbstdefinierte

Konzeptionell

- Einführung von Mechanismen und Datenstrukturen, die den funktionalen SOLL Zustand ermöglichen
- Bessere programmtechnische Erweiterbarkeit

Vortrag von
2015

- Anwendungsschnittstellen „vollständig“ (Server/Client)
- Umfangreiches Client-Framework „KISS“ „fertig“, 100% TypeScript

Oberfläche

- Durchgängiges Bedienkonzept
- Neue GUI Komponenten (Core-Framework)
- Zusammenführen verschiedener Werkzeuge
- X-Browser fähig

- Bedienkonzept wahrscheinlich gelungen
- Design: Kritik ... angekommen, in Umsetzung
- X-Browser gegeben

Serverseitig

- Vollständige Ablösung von classic ASP
- Volle 64Bit Fähigkeit
- Komplett kompatibel mit cardo3 Backend
- Sitzungslos

Vortrag von
2015

- Alle Punkte erfüllt
.... Aber: z.Z. Auslieferung noch als 32 Bit Version

Für Dritte

- Leistungsfähigere API
- Einfache und sichere Nutzung der API (Dokumentation, Unterstützung durch Entwicklungsumgebung)

Vortrag von
2015

- Platz für eigene Meinung: Siehe Beispiel im Laufe des Vortrages

Entwicklungsplan

- Fertigstellung Bedienkonzept und Layout
- Migration weiterer Basisfunktionen
- Schaffung Ausführungsschicht für cardo3 Anwendungen
- Umbau ausgewählter Fachanwendungen auf neue Technologie
- Abschließender Umbau aller Fachanwendung auf neue Technologie

- Weitestgehend ...

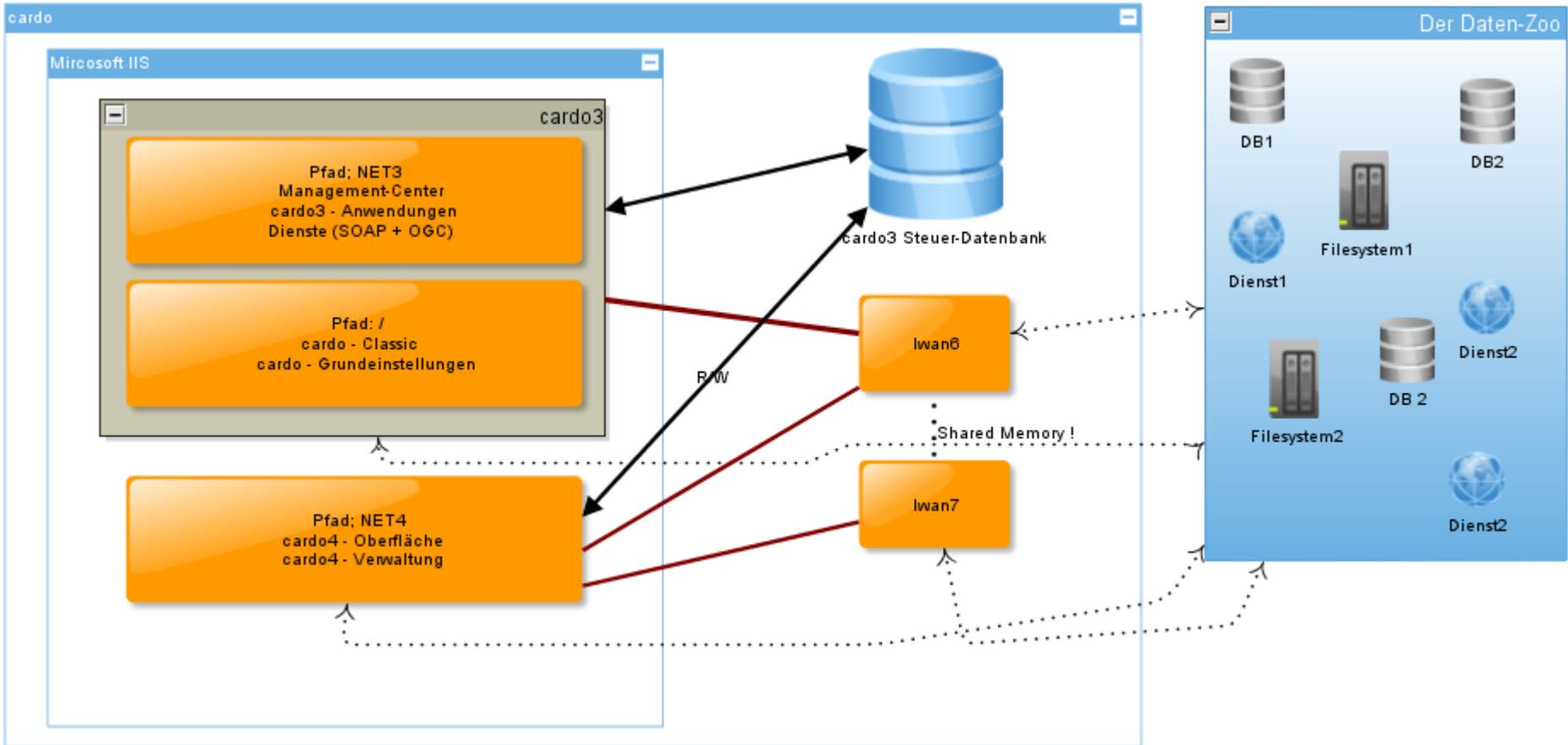
Zeitplan

- Erste Auslieferungen an Neukunden bzw. für bestimmte Projekte im I Quartal 2016
- Integration der cardo3 Anwendungsschicht ca. Mitte 2016
- Neuerstellung ausgewählter Anwendungen ??
- ...
- Alles wird cardo4

- Mit Blick auf den Kalender ... passt

Migration

- Installation besteht aus dem Einrichten eines virtuellen Verzeichnis „net4“ im IIS
- (fast) alle Einstellungen werden aus dem cardo3 Managementcenter verwendet (!)
- Aktualisierung über „IDUUpdate“
- Einige wenige Einstellungen sind in cardo4 vorzunehmen



Server

- Mind. Server Version: 2008R2, empfohlen 2012R2
- .Net Version 4.6.1

Client

- Mind. Browser-Version: IE11 (oder alle anderen neueren Browser)

- Parallelbetrieb cardo3 / cardo4 möglich
- Empfehlung: Schrittweise Einführung

Hausinterne Aufwände:

- Schulung der Betreuer (geringfügig)
- Schulung der Mitarbeiter
- Ggf. Anpassungen an eigenen Anwendungen
- Ggf. Entwicklerschulungen

Finanzielle Aufwände

- Lizenz-Upgrade
- Anpassungen vorhandener cardo3 Anwendungen (rel. geringer Aufwand)
- Neuerstellung von cardo2 Anwendungen („classic ASP“)

... wenn diese in cardo4 benötigt werden

- Upgrade Lizenz für cardo4
- Kosten fallen je Kunde an, nicht je Installation
- Keine Erhöhung der bestehenden Wartungskosten
- Für in der folgenden Folie nicht aufgelistete Kern-Module fallen keine weiteren Kosten an

- Auf Anfrage unter support@idu.de

Mehr & Weniger

Stand

Themenbaum

- Ja, es gibt ihn noch

Handlungsbedarf:

- Keiner

Navigation in der Karte

- Besser, „Schalterlose“ Bedienung, wie cardo.Map

Handlungsbedarf:

- Keiner

Selektion/Auswahl

- Umfangreiche Sach- & Geometrierecherche
- GeoSQL (später mehr)
- MapTip (Erfahrung: kann oft die Selektion ersetzen)

Handlungsbedarf:

- Selektionen im Sinne der Kartendarstellung
- Einfache Selektion per Geometrie
- Verschneidungstool (Betroffenheitsrecherche)

Messen / Zeichnen

- Messfunktion inkl. Beschriftung Ausdruck

Handlungsbedarf:

- Sketch für Zeichnungen
- Shape-Export der Redline

Nutzerbezogene Kartendarstellungen

- Reihenfolge anpassen, Sichtbarkeit etc. im Gis-Viewer
- Für Iwan7 Ebenen implementiert (später mehr)

Handlungsbedarf:

- ‚Symbolik anpassen‘ - Dialog
- ‚Beschriftung erstellen‘ - Dialog
 - Einstellungen in Sitzung speichern
 - Überlegung: versch. Darstellung in versch. Viewern

Drucken

- Implementierung, analog cardo3

Handlungsbedarf:

- Stapel-Druck
- Print-Designer in UI integrieren

Export

- Excel Export für alle Daten-Module implementiert

Handlungsbedarf:

- Bild in Zwischenablage
- Geodaten-Export in UI integrieren

Personalisierung

- Weitaus umfangreicher als bisher, stärker konsolidiert

Handlungsbedarf:

- Aufsplitten der Sitzungseinstellung in Bestandteile
- Freigabe von Sitzungen und anderen Dingen an Benutzerkreise

Upload

- Fehlt gerade wieder

Handlungsbedarf:

- UI neu implementieren, Nutzung neuer Möglichkeiten, wie Datei Drag&Drop
- Anbindung an Freigabemechanismus

GSS / Gazetteer / Koordinaten-Tool /
Übersichtskarte / cardo.COM

- O.g. sind verfügbar
- Gazetteer ist in Suchfeld integriert

Handlungsbedarf:

- keiner

Weitere

- Gedit und gKK Bearbeitung sind integriert
- Metadaten (Editor, Anzeige, Recherche) dito

Handlungsbedarf:

- Weitere Cardo3 Anwendung integrieren (z.B. ALKIS, Puzzle, Button) => Realisierung über cardo3 „Wrapper“, Nur Anpassungen an Client-Schnittstelle erforderlich
- gKK: Verwaltungsanwendung neu erstellen
- ToPG: Pendant neu erstellen

Keine der „Backendanwendungen“ erfordert Anpassungen, diese sind vollumfänglich nutzbar!

- c2c, lxx Export/Import,
- ALKIS Konverter, Xplan Konverter,
- GeoTools
- etc. ...

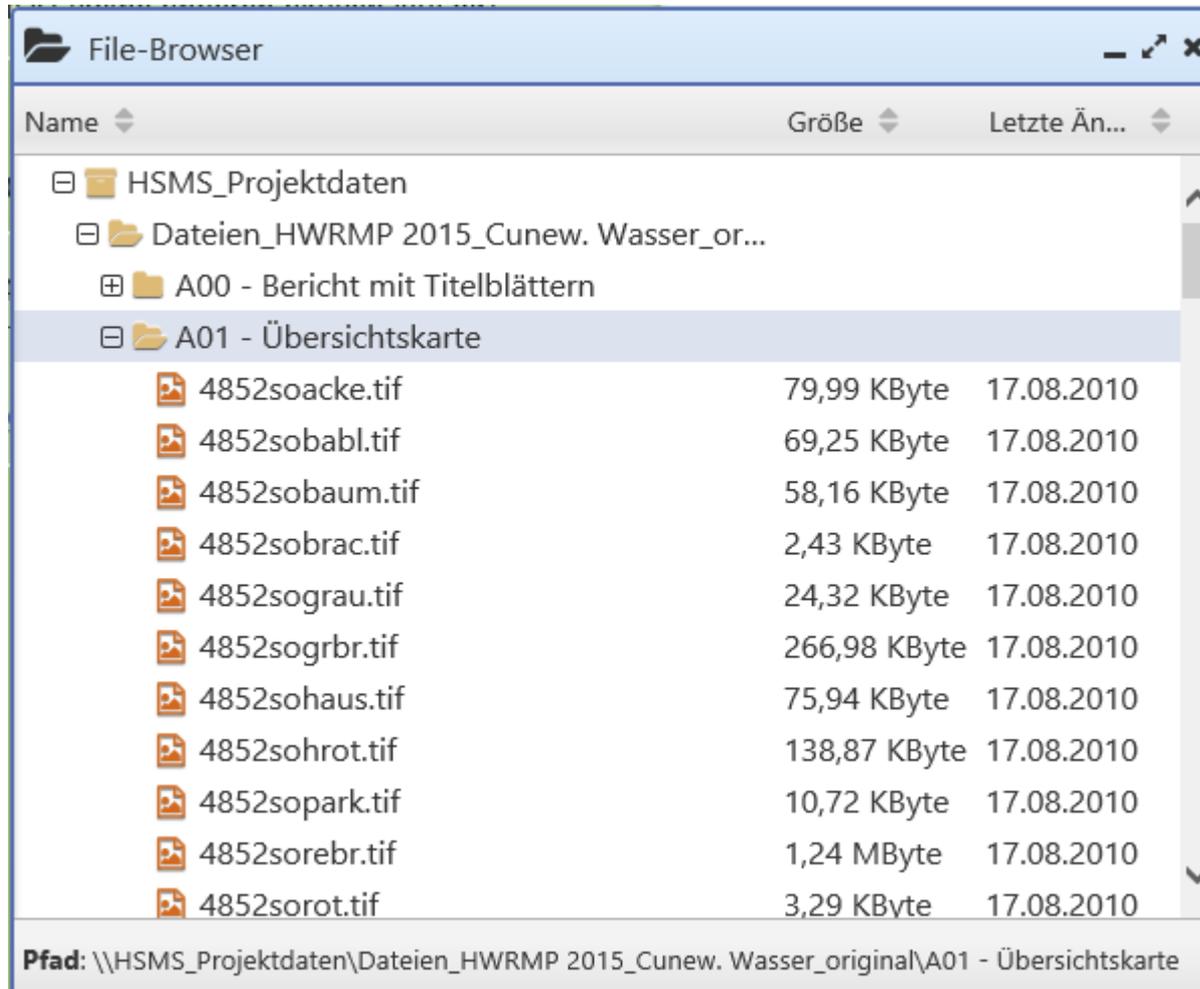
Mehr

Überblick über die (neuen) Module in cardo4

* im Basispaket enthalten

Funktionen

- Einfache Ablage von Dokumenten aller Art im Filesystem
- Berechtigungen pro „Laufwerk“
- Aus Anwendungen heraus aufrufbar
- Download von Dokumenten oder Ordnern
- Spezielle Behandlung von Quelldateien mit Geodaten



Ideen

- Alternative zum Admin-Tree für „lose“ Sammlungen von Geodaten
- Bereitstellung projektbezogener unstrukturierter Daten
- Evtl. in Kombination mit anderen Tools, z.B. Speicherung der Metadaten in cardo.Puzzle => Link auf Ordner

Funktionen

- „SQL für alles“
- Abfragen für alle Vektordatenquellen
- SQL92 kompatible Syntax
- Viele SQL SFS Funktionen (ST_xx)
- Datenquellenübergreifende Abfragen
- Meint sowohl API als auch UI
- Prinzip bedingt unanfällig gegenüber SQL Injection

GeoSQL LT

Ausführen Excel Export

```

1 -----
2 -- Gemeinde (430 Datensätze)
3 -- Shape-Datei
4 -----
5 SELECT
6 *
7 FROM
8   L416
9 LIMIT 10
  
```

Shap...	ShapeGeo...	SCHL...	ORTS...	KREIS	ADMIN	SITZ_...
0	Geometrie	14523010	Adorf/V...	Vogtlan...	Stadt	N
1	Geometrie	14628010	Altenber...	Sächsisc...	Stadt	n
2	Geometrie	14522010	Altmittw...	Mittelsa...	Gemeinde	N
3	Geometrie	14521010	Amtsberg	Erzgebir...	Gemeinde	N
4	Geometrie	14521020	Annaber...	Erzgebir...	Große Kr...	J
5	Geometrie	14625010	Arnsdorf	Bautzen	Gemeinde	N
6	Geometrie	14730010	Arzberg	Nordsac	Gemeinde	N

```

});
if (Kiss.Lang.String.IsNullOrEmpty(cfg.strassenLayerName))
{
    let sql = `
SELECT DISTINCT
    name_prim as id,
    name_prim as title
FROM
    ${strLayerName} as STR,
    (SELECT geom FROM ${gemeindeLayerName} where ${gemeindeColName} in (${filterWords})) as ge
WHERE
    str.geom match gem.geom and ST_Intersects(str.geom, gem.geom)`;
return CorrectionsApplicationRemote.AxGetLayerDataByGeoSql(sql)
    .then(table =>
    {
        let idIdx = IduIT.Core.Kiss.Lang.Array.first(table.columns, (c) => IduIT.Core.Kiss.Lang.String.Equals(c, 'id'));
        let nameIdx = IduIT.Core.Kiss.Lang.Array.first(table.columns, (c) => IduIT.Core.Kiss.Lang.String.Equals(c, 'title'));
        //let idColIdx = table.columns[idIdx];
        result.categories.push(...table.data.map(row =>
        {
            return {
                categoryId: row[idIdx],
                categoryTitle: row[nameIdx]
            };
        }));
        return result;
    });
});
}
else
{
    return result;
}
  
```

Anforderungen an den Nutzer:

- SQL Kenntnisse erforderlich
- Detaillierte Kenntnis der Semantik der Daten

Nicht spannend:

SELECT

*

FROM

L416

Auch nicht spannend:

SELECT

*

FROM

L1

WHERE

ID = 5

Interessanter:

SELECT

kreis,

sum(ST_Area(ShapeGeometry))

FROM

L1

GROUP BY Kreis

Und wie ist es mit komplexeren Fragestellungen?

„Liegt das Flurstück 1/13 der Gemarkung Bautzen in oder an einem Gewässer? Zeige mir die fünf dem Flurstück am nächsten liegenden Gewässer sowie deren Abstand zum Flurstück an.“

Interessanter:

SELECT

```

gewaesser.gewkz AS "GewaesserKennzahl",
gewaesser.name AS "Name",
gewaesser.geom AS "Gewässer",
flst.geom AS "Flurstück",
round(ST_Distance(flst.geom, gewaesser.geom)) AS "Abstand [m]"

```

FROM

```
L276 AS gewaesser
```

```
INNER JOIN L282 flst ON match(st_buffer(flst.geom, 50), gewaesser.geom)
```

```
WHERE flst.gemarkung = 'Auritz'
```

```
AND flst.fbz = '5/4'
```

```
AND round(ST_Distance(flst.geom, gewaesser.geom)) <= 50
```

ORDER BY

```
ST_Distance(flst.geom, gewaesser.geom) ASC LIMIT 5
```

z.Z. abfragbare Datenquellen

- WFS Dienste (1.1, 1.0, 2.0)
- PostgreSQL
- MS SQL Server
- Oracle
- ODBC (Punkte)
- ShapeFile

Funktionen

- Parametrisierbare Oberfläche für GeoSQL
- Vorgehen: Betreuer stellt Abfragen zusammen, Nutzer bekommt einfaches Formular mit (optional) Parametern

GeoSQL
? - ↗ ✕

+ Abfrage hinzufügen

- [-] **Grundwasser**
- Gewässereinzugsgebiete
- Trinkwasserschutzgebiete
- Trinkwasserschutzgebiete im LK
- Überschwemmungsgebiete
- Überschwemmungsgebiete im LK
- [-] **Hochwassermarken**
- Hochwassermarken im Landreis
- Hochwassermarken in einer Gemeinde
- [-] **Oberflächenwasser**
- Abstand zum Gewässer
- Betroffenheitsrecherche
- Betroffenheitsrecherche - Anzahl Gebiete
- Einleitstellen
- Einleitstellen im Landkreis
- Gewässer in einer Gemeinde
- Gewässerlängen für eine Gemeinde
- Gewässerlängen im Landkreis

Abstand zum Gewässer

Liegt das Flurstück x/y der Gemarkung x/y in/an einem Gewässer?
 Angezeigt werden die fünf dem Flurstück am nächsten liegenden Gewässer sowie deren Abstand zum Flurstück.
 Wird kein Datensatz ausgegeben, bedeutet das, dass sich im Umkreis des maximalen Abstands kein Gewässer befindet.

Parameter der Abfrage (jeder Parameter ist mit einem Wert zu belegen)

Maximaler Abstand:

Filter der Abfrage (nicht optionale Filter sind mit einem * gekennzeichnet)

Gemarkung*:

Groß-/Kleinschreibung beachten

Flurstück*:

Groß-/Kleinschreibung beachten

Ausblick

- Verschmelzen mit cardo.Button => GeoSQL als weiterer Dateninput
- Laufender Ausbau der Datenquellen ... nach Bedarf
- Anpassungen weiterer SQL Funktionen ... nach Bedarf
- Persistenz von Abfragen (CREATE TABLE AS ...)

Funktion

- Eigentlich noch keine
- Am besten zeigen! (mit einem anderen Anwender)

Ideen

- Zusammenarbeit für Kollegen untereinander (UI)
- Benachrichtigungs-System für Anwendungen (API)
(Datensatz X wurde von Person Y geändert,)

Iwan7

- eigenständiger HTTP Server (wie bisher)
- Windows Dienst (wie bisher)
- 64 Bit Version (bisher nur 32Bit Produktiv)
- Ebenen Definition per JSON (kein Iws mehr)
- Interne Ebenen-Typen:
 - Vektor
 - Raster
 - CAD

- Trennung zwischen Quellendefinition und Darstellung (aber Standard-Stil vorhanden)
- Zeichen-Vorgang: Trennung zwischen Datenbeschaffung und eigentlichem Rendering
- Abfragen: einfache (XML, wie bisher) oder komplexe über GeoSQL
- Zugriff über .NET Schnittstellen (wie bisher)
- Detailliertere Datentypen (Int4, Int8, Objekte (!), ...)

- Beschreibung über eine an CSS angelehnte Syntax
- Alle Ausdrücke können als Literal oder „Expression“ angegeben werden
- Neue Ideen und Features, z.B. Trennung
Flächenumgrenzung Außen/Innenring, alle Farben als
ARGB
- Komplexe Muster für alle Typen definierbar
- Stil kann pro Zeichenvorgang übergeben werden
- Anwendbar auf Raster und Vektorebenen
- Siehe auch <http://www.cardogis.com/?pgId=978>

```
unordered::line::Textausrichtung [mapscale<4000] {  
    line-offset: 1m;  
    text: [txt];  
    text-color: black;  
    text-font-name: "Arial";  
    text-height: 3m;  
    text-weight: 400;  
    text-quality: antialiased;  
}
```

```

special-raster-properties [mapscale > 0]
{
  /*die Transparenz der Tiff-Dateien*/
  image-opacity: 1.0;
  /*Kanal-Wahl*/
  red-channel-index: 0; green-channel-index: 1; blue-channel-index: 2;
  one-bit-color: black;
  /*eine Liste mit Farben die 100% transparent dargestellt werden sollen*/
  transparent-colors: RGB(0, 255, 0), #ffff00ff, red, ARGB(255, 255, 255, 255);
  /*Grauwertbild ?*/
  render-gray-scale-image: false;
  /*Invertierte Darstellung*/
  invert-colors: false;
  render-quality: high;
  /*Color-Matrix ...
  color-matrix-red-from-rgbat: 0.393, 0.769, 0.189, 0.0, 0.0;
  ...
  */
}

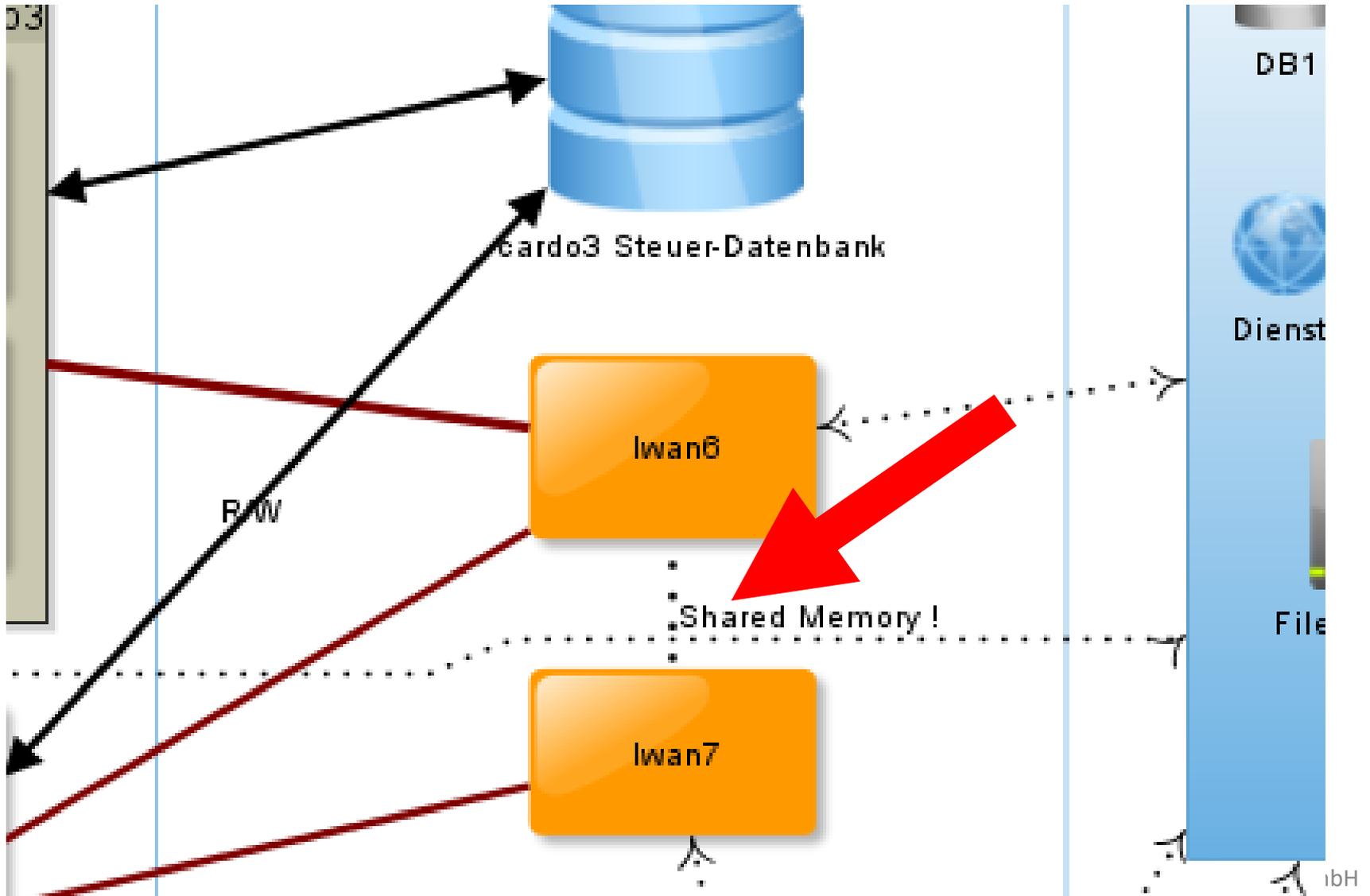
```

Expressions:

- Zugriff auf alle Spalten der Datenquelle und einige Konstanten (z.B. aktueller Maßstab)
- Verwendung als Filter oder für Wertebestimmung
- Numerische Berechnungen und Zeichenketten Operationen möglich
- Bsp.:
[5 + ((mapscale < 1000) ? 10 : 20)]
[SPALTE * 2]

- PostgreSQL / Postgis
- Microsoft SQLServer
- Oracle SDO
- ODBC (Rechts/Hochwerte)
- ESRI Shapefile
- DXF
- RasterLite2
- TIFF (tfw und GeoTIFF), mit Support für Rotation etc.
- WFS (1.0, 1.1, 2.0)
- WMTS
- OpenStreetMap
- EMF

- Produktiv nutzbar
- Einbindung über cardo3 bereits integriert (seit 3.8.5)
- Brücke z.Z. Iwan6 greift auf Iwan7 zu
- Fehlende Features:
 - Abfragen über Iwan6
 - Generierung von Legendenbildern



Kurzfristig in Planung:

- Bereitstellung Legendensymbole
- Erweiterung der Ebenentypen, (z.B. WMS, PDF, GML, nach Bedarf)
- Ausbau CSS / Rendering

Langfristiges Ziel:

- Ablösung von Iwan6, Nutzung aller neuen Features in cardo4

If ... then ... else

Warnhinweis:

Die folgenden Folien enthalten
Spuren von Programmcode!

ckeln mit cardo4

Cardo - als Entwicklungs-Framework

- Monolithische Systeme gibt es viele – cardo kann hier mehr
- Erweiterung war und ist ein Grundkonzept des Systems

Zu betrachtende Schnittstellen

Serverseitig:

- Anwendungen
- Suchprovider

Clientseitig:

- Anwendungen
- Erweiterung bestehender Module (PlugIn)

Cardo.COM:

- Für Dritte, browserbasierte-Mechanismen (Postmessage oder Url-basiert)

Einzusetzende Technologien:

Serverseitig:

- C# / .Net 4.6

Clientseitig:

- TypeScript 1.7 (bald 2.0)

Entwicklungsumgebung:

- Notepad
- Oder ... Visual Studio 2015 (z.Z. mit max. SP1!), empfohlen

- Alle Erweiterungen werden nur noch per Typ-Schnittstelle ermittelt (d.h. eine DLL muss vorhanden sein)
- Schnittstellen mit den Erfahrungen der letzten Jahre konsolidiert und vereinfacht
- Clientseitig enorme Verbesserung durch TypeScript und die umfangreiche, tatsächlich dokumentierte, vollständig neue API und KISS

Schnittstelle:

`cardo.Core.Api.ISearchProvider`

Zweck:

Erweiterung der durchsuchbaren Inhalte in *dem* Suchfeld

Methoden:

```
IEnumerable<SearchMatch> Search(HashSet<string> terms, string term)  
SearchMatchBehavior MatchBehavior {get;}
```

```

internal sealed class BeispielSuche : cardo.Core.Api.ISearchProvider
{
    public IEnumerable<SearchMatch> Search(HashSet<string> terms, string term)
    {
        if ((terms.Contains("anwendertreffen")
            && terms.Contains("wo")))
        {
            yield return new SearchMatch(new WellKnownType.FeatureCollection()
            {
                Columns = new[] { "Title" },
                title = "POI",
                Values = new[] {new FeatureCollection.Feature()
                {
                    Geom=Geometry.CreatePointGeometry(413122,5662537,
                        IDU.GeoLib.Epsg.AllEpsgCodes[25833]),
                    Data = new object[] {"CAT 2016"}
                }}
            });
        }
    }
    public SearchMatchBehavior MatchBehavior
    {
        get { return SearchMatchBehavior.StopFurtherSearchesIfNotEmpty; }
    }
}

```

- Hinweis: Nutzerdefinierte Suchprovider werden vor den integrierten ausgeführt

Schnittstelle:

ICardoManagedApplication

Zweck:

Erstellung einer eigenen Anwendung, die von cardo verwaltet wird
(Berechtigungen, SQL Updates, Einstellungen etc.)

Methoden:

```
String InitOnlyTitle { get; }
```

Schnittstelle:

IClientApplicationDefinition

Zweck:

Erstellung einer eigenen Anwendung, die eine Oberfläche hat

Methoden:

```
String InitOnlyTitle { get; }  
void RegisterDefinitionScripts(ResourceCollector resourceCollector);  
void RegisterApplicationScripts(ResourceCollector resourceCollector);
```

Wenn genügend Zeit ist, dann erstellen wir jetzt
gemeinsam eine Anwendung!

- Fazit:
 - Alles Gute und seit Jahren Bewährte aus cardo3 beibehalten (umfangreiche Berechtigungen, Verwaltung, etc.)
 - Alles andere Richtiger gemacht:
 - Sehr aufgeräumte serverseitige Schnittstellen
 - Clientseitige Schnittstellen zu allen Aspekten der Umgebung
 - Integration der Ajax Umgebung
 - Visual Studio Extension für maximale Produktivität
 - Es gibt vieles in „Grün“

cardo4 Anwendungen

Schnellstart

 Karte

 Lärmkartierung

Lärmkartierung ? [] [x]

Über
 Legende&Karteninhalt
 Mein Gebiet
 Meine Korrekturen
 Verwaltung

+ Einen neuen Eintrag erstellen

Daten

Zuordnung

A17 v x

Hinweis

Gebäude fehlt

B 172

Gebäude 172
1 themenspez. Hinweis.
Erstellt am 25.10.2016 um 15:50:49 Uhr

B 172A

Straßen s
3 themenspez. Hinweise, 1 Geometrien.
Erstellt am 13.10.2016 um 15:12:29 Uhr

Bauarbeiten
2 themenspez. Hinweise.
Erstellt am 18.10.2016 um 09:19:29 Uhr

ohne Kategorie

Test
2 themenspez. Hinweise.
Erstellt am 13.10.2016 um 15:25:56 Uhr

S 167

Lärmschutz
2 themenspez. Hinweise.
Erstellt am 14.10.2016 um 10:24:32 Uhr

Hochhäuser
2 themenspez. Hinweise.
Erstellt am 24.10.2016 um 10:28:06 Uhr

S 168

Gebäude am Park
Lage
Erstellt am 24.10.2016 um 10:29:35 Uhr

SX
1 themenspez. Hinweis.
Erstellt am 25.10.2016 um 13:28:09 Uhr

Themenspezifische Einträge

Ihre Ergänzung im Thema **Gebäude**

Lage

Allgemeiner Hinweis

Ihre weiteren Angaben zum fehlenden Eintrag

Rel. Höhe (m)

Funktion/Nutzung

Speichern Entfernen

Ihre Ergänzung im Thema **Gebäude**

Lage

Allgemeiner Hinweis

Ihre weiteren Angaben zum fehlenden Eintrag

Rel. Höhe (m)

Funktion/Nutzung

Speichern Entfernen

[+ Anfügen](#)

Geometrien

Sie können hier weitere Geometrien mit Hinweisen hinzufügen. Korrekturen an bestehenden Objekten sollten Sie besser dem entsprechendem Thema zuordnen.

Vergessen Sie nicht, nach Änderungen an einzelnen Einträgen diese zu "Speichern".

Karte ? [] [x]

Messwerkzeug

EPSG:25833 1:804

Schnellstart

Querprofile Gewässerabschn... WRRL - Import Karte

Querprofile

+ Neues Querprofil erstellen

- Bodenbach
 - Station 1086
 - Station 1180
- Butterwasser
- Grenzbach Berge
- Hoyerswerdaer Schwarzwasser
- Spree

Station 1180

Bemerkung

Sohlgefälle (m/m) 0,08818

Rauigkeit (k_{90}) 30

Benetzter Umfang (m) 2,36

Abflussquerschnitt (m²) 0,41

Hydraulischer Radius (m) 0,17

Fließgeschwindigkeit (m/s) 2,77

Bordvoll-Abfluss (m³/s) 1,14

Gewässerabschnitte

kleine spree

- Bärhäuser Wasser
- Baruther Flutgraben
- Baruther Flutgraben
- Baschützer Wasser
- Binnewitzer Wasser
- Bodenbach
 - Alternative Gewässernamen
 - Gewässerstrukturgütekartierung
 - Gewässerordnung
 - 0 - 1193
 - Hydraulische Leistungsfähigkeit
 - Bohraer Wasser
 - Bohraer Wasser; Luggraben
 - Bolbritzer Wasser
 - Borngraben
 - Briesinger Graben
 - Bröthener Mühlgraben
 - Buchholzer Wasser
 - Burkauer Wasser
 - Büschingsgraben
 - Büschingsgraben; NN
 - Buschwasser
 - Buttermilchwasser

Ordnung

Querschnittsgeometrie

6m -4m -2m 0m 2m 4m

2m

1m

0m

-1m

-2m

Verschieben > < Abbrechen Übernehmen

WRRL - Import

Anzahl vorh. Maßnahmen: 573

letzter Import am: 25.10.2016 13:30

letzter Import durch: lkbautzen

Karte

Ebenen

Grundkarte

OpenStreetMap

Ebenen-Reihenfolge

Im Vordergrund

cardo-Ebenen

Gewässernetz Topol...

Grundkarte

Im Hintergrund

Schnellstart

DGN/DXF Datei... WIH Steuerung Karte

DGN/DXF Dateitool

Datei

Fügen Sie Dateien durch Klick auf "Gebiet hinzufügen" oder durch Aktionen in der Karte hinzu.
Beachten Sie, dass in der Karte Themen aktiv sein müssen, die auf Plandokumente verweisen (Entwurfsvermessung, Stadtkarte R-Teil, Vermessungen wo DXF da sind, Stadtkarte G-Teil).

Daten in CAD-Anwendung öffnen

WIH Steuerung

Weiche	Stellsys.	Standort	Örtlichkeit
1	HW61	Hamburger/ Waltherst...	Hamburger Straße
2	HW61	Straßburger Platz	Canalettostraße
3	HW61	Straßburger Platz	Canalettostraße
4	HW61	Straßburger Platz	Stübelallee
5	HW61	Am	
6	HW61	Gle	
7	HW61	Kör	
8	HW61	Ant	
9	HW61	Pos	
10	HW61	Pos	

Karte

Messwerkzeug

Leistungsarten

Weiche: 4 Ort: Straßburger Platz

Leistungsart	Startt...	Frist	zykl...	entf...	entf...	Prüf...	letzte...	nächst...
Grundreinigung	22.07....	364	ja			ja		22.07....
Reinigung	01.07....	3.5	ja			ja		21.12....
Spannkeilke...	02.04....	364	ja			ja		02.04....
Grundreinigung		364	nein	09.05....		nein		22.07....
Reinigung			nein			nein		21.12....
Reinigung - S			nein			nein		23.12....
Reinigung - S		90	nein			ja		23.12....
Reinigung - Z		1	nein			nein		04.12....

Schließen