



FIZ Karlsruhe – Leibniz-Institut für Informationsinfrastruktur

4. RADAR-WORKSHOP KARLSRUHE, 25./26. JUNI 2018

RADAR APPLICATION PROGRAMMING INTERFACE

Matthias Razum, FIZ Karlsruhe

APPLICATION PROGRAMMING INTERFACE

POTENZIAL

ANWENDUNG




```
...e[b]())}}var c=function(b){this.element=a(b),c...
opdown-menu"),d=b.data("target");if(d||(d=b.attr("href"),d=d&&d.replace(/.*(?:#\w+)?/,""),d...
st a"),f=a.Event("hide.bs.tab",{relatedTarget:b[0]}),g=a.Event("show.bs.tab",{relatedTarget:e[0]}),...
faultPrevented()){var h=a(d);this.activate(b.closest("li"),c),this.activate(h,h.parent(),function...
rigger({type:"shown.bs.tab",relatedTarget:e[0]}))}}},c.prototype.activate=function(b,d,e){func...
> .active").removeClass("active").end().find('[data-toggle="tab"]').attr("aria-expanded",!1),...
ia-expanded",!0),h?(b[0].offsetWidth,b.addClass("in")):b.removeClass("fade"),b.parent(".dropdo...
().find('[data-toggle="tab"]').attr("aria-expanded",!0),e&&e()}var g=d.find("> .active"),h=e&&...
le")||!!d.find("> .fade").length);g.length&&h?g.one("bsTransitionEnd",f).emulateTransitionEnd...
;var d=a.fn.tab;a.fn.tab=b,a.fn.tab.Constructor=c,a.fn.tab.noConflict=function(){return a.fn.tab...
show"));a(document).on("click.bs.tab.data-api",[data-toggle="tab"],e).on("click.bs.tab.data...
se strict";function b(b){return this.each(function(){var d=a(this),e=d.data("bs.affix"),f="ob...
typeof b&&e[b]())}}var c=function(b,d){this.options=a.extend({},c.DEFAULTS,d),this.$target=a...
,a.proxy(this.checkPosition,this)).on("click.bs.affix.data-api",a.proxy(this.checkPositionWi...
null,this.pinnedOffset=null,this.checkPosition());c.VERSION="3.3.7",c.RESET="affix affix-top...
State=function(a,b,c,d){var e=this.$target.scrollTop(),f=this.$element.offset(),g=this.$targ...
"bottom"==this.affixed)return null!=c?!(e+this.unpin<=f.top)&&"bottom":!(e+g<=a-d)&&"bottom"...
!=c&&e<=c?"top":null!=d&&i+j>=a-d&&"bottom"},c.prototype.getPinnedOffset=function...
.RESET).addClass("affix");var a=this.$target.scrollTop(),h=this...
withEventLoop=function(){setTimeout(a,n)};...
nt.height(),d=this
```

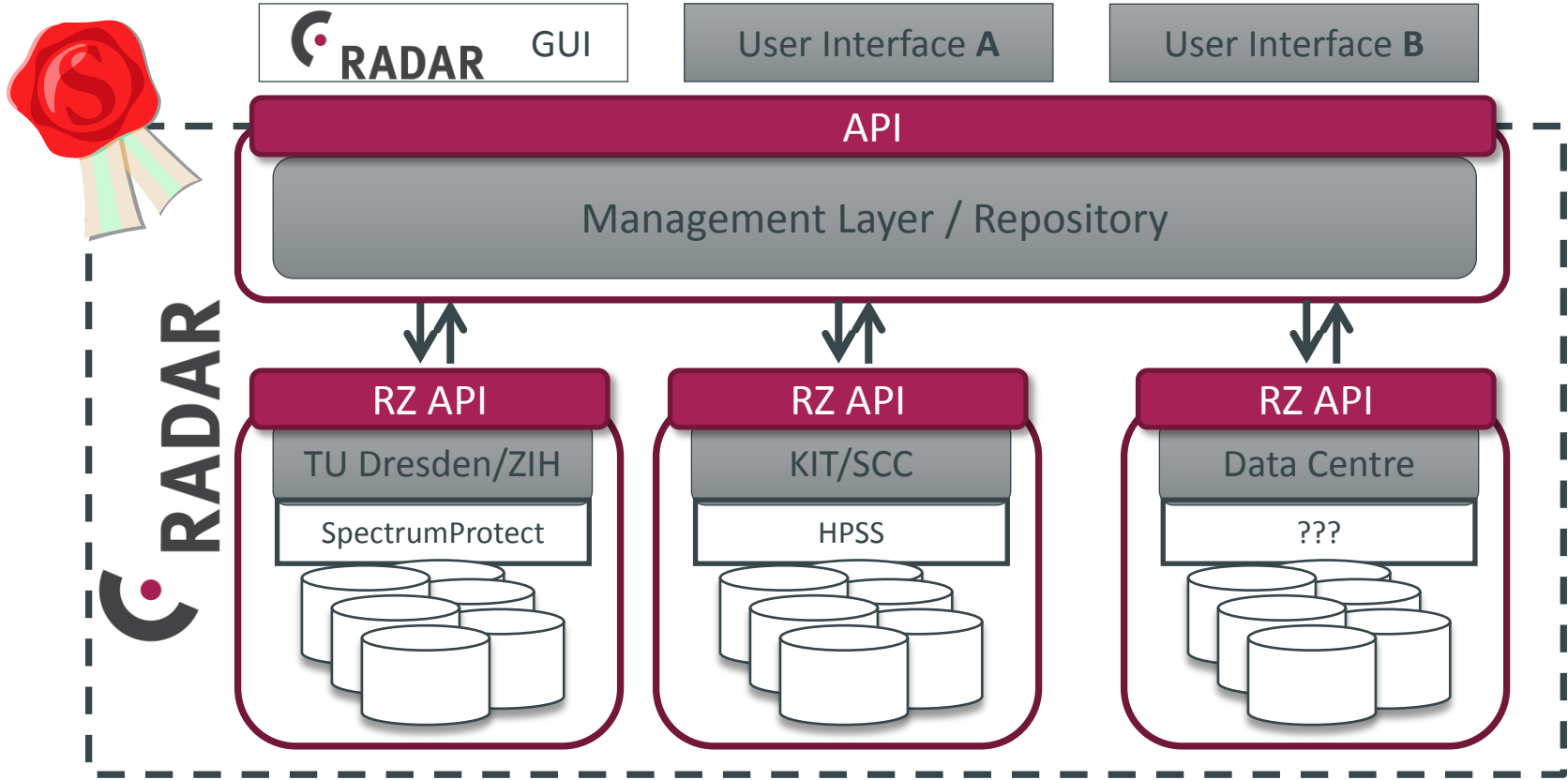

Application Programming Interface

Oder deutsch: Programmierschnittstelle

„... is a set of subroutine definitions, protocols, and tools for building application software. In general terms, it is a set of clearly defined methods of communication between various software components.”

Quelle: Wikipedia, CC BY-SA 3.0

Schematische Architektur



APPLICATION PROGRAMMING INTERFACE
POTENZIAL
ANWENDUNG



„The coolest thing to do
with your data
will be thought by someone else.“

Die JISC Common Repository Interfaces Group (CRIG)

Wofür kann das API genutzt werden

- Einbindung von RADAR in eigene Workflows
 - z.B. Automatisierter Upload von Daten nach RADAR
- Ersatz des RADAR-Frontends durch eigenes System
 - Disziplinspezifische Angebote mit abweichenden Metadaten-Schemata
- Integration von RADAR-Datenpaketen in institutionelle Suche
 - Auch ohne Programmierung über OAI-PMH
- Parallele Nutzung von RADAR-Frontend und eigenem Code möglich

APPLICATION PROGRAMMING INTERFACE
POTENZIAL
ANWENDUNG



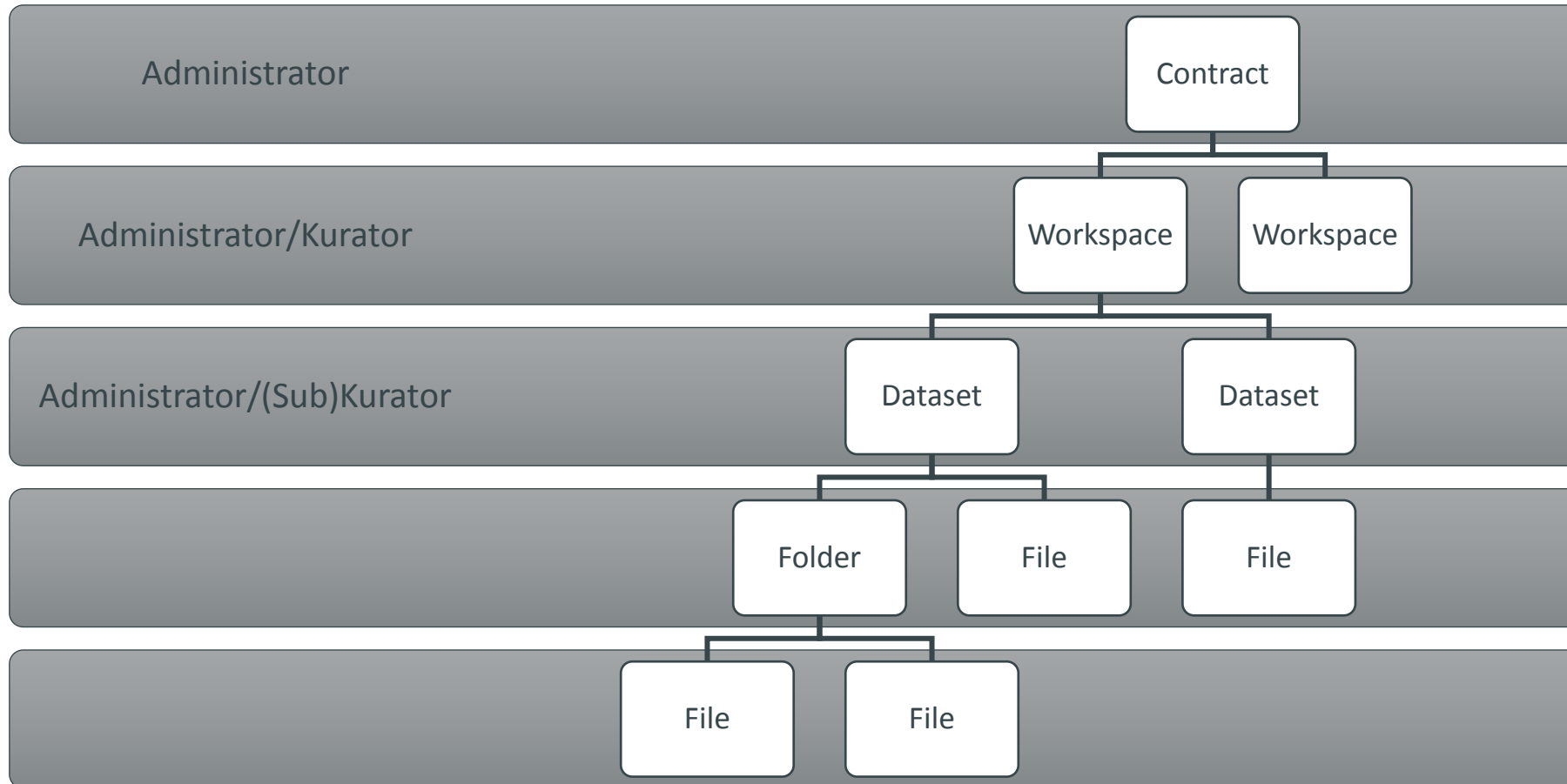
Voraussetzungen

- HTTP Client
 - cURL, Wget, Firefox, Java, Python
- Zugriff auf das RADAR API über das Internet
 - Firewall, Proxy-Server etc.
- Zugangsdaten für OAuth Client
 - Kann über info@radar-service.eu angefordert werden
- Gültiger RADAR-Benutzer
 - Über die Registrierung selbst anlegen
- Notwendige Zugriffsrechte auf Vertrag, Arbeitsbereiche und Datenpakete
 - Rechtezuweisung durch lokalen Administrator

Sicherheit und Authentifizierung

- Das API ist über das OAuth-2.0-Protokoll abgesichert
 - Standard für verteilte Authentifizierung im Web
 - RADAR ist in sich bereits ein verteiltes System mit mehreren Servern
- RADAR Authentication Server (OAuth Resource Server)
 - Programm authentifiziert sich gegen RADAR Authentication Server im Namen eines Benutzers
 - Es erhält ein Token
 - Zugriff auf Methoden des RADAR-APIs nur mit gültigem Token
- Die Credentials des Benutzers werden nur an den RADAR Authentication Server übertragen

Hierarchische Organisation der Inhalte



Rollen und Rechte

Administrator

- Verwaltet Arbeitsbereiche
- Vergibt Rollen für Administratoren, Kuratoren und Subkuratoren
- Hat Zugriff auf Statistiken und verwaltet Quotas

Kurator

- Volle Schreib- und Leseberechtigung für Arbeitsbereich
- Kann Datenpakete publizieren/archivieren
- Kann Review-Prozesse anstoßen und beenden
- Vergibt Rollen für Kuratoren und Subkuratoren

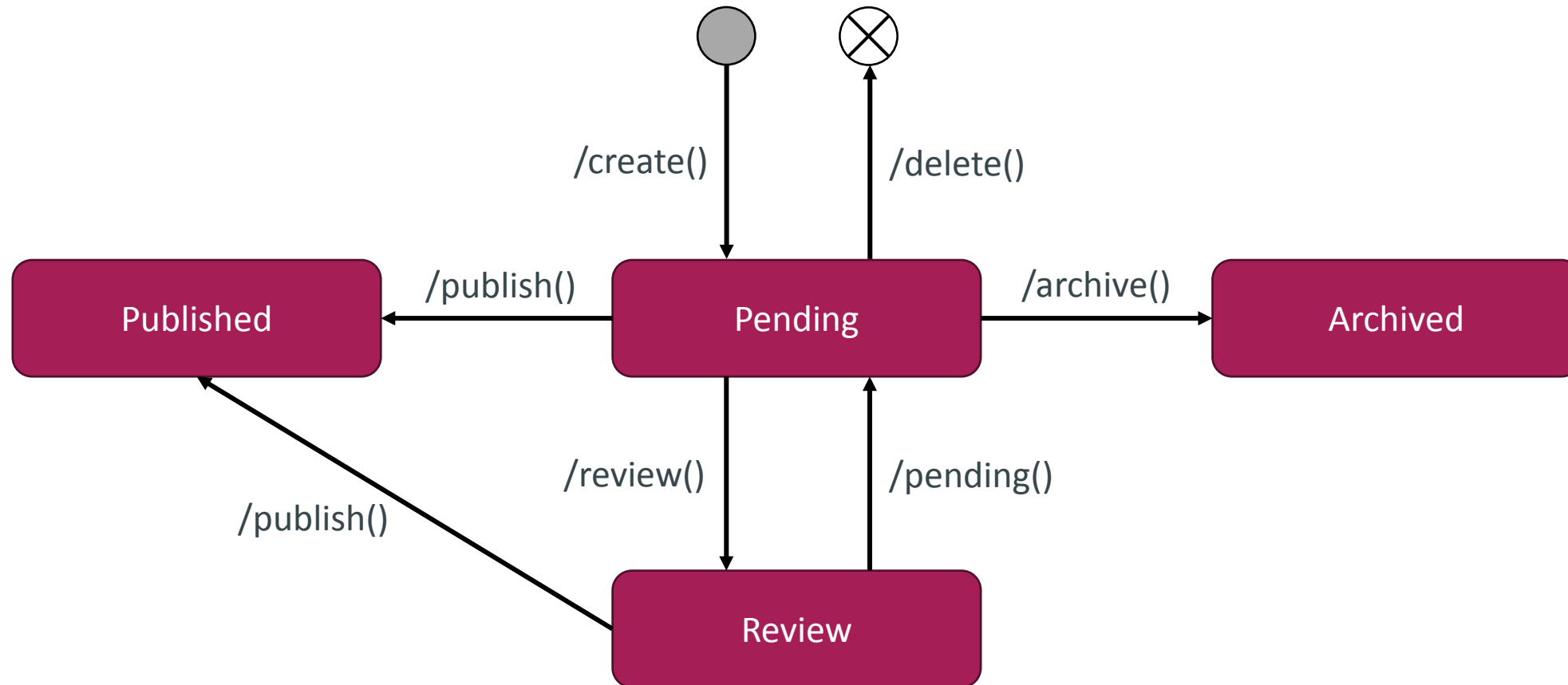
Subkurator

- Volle Schreib- und Leseberechtigung für Arbeitsbereich
- Kann keine kostenpflichtigen Aktionen auslösen (keine Archivierung oder Publikation)

Dataset User

- Erfordert Authentifizierung
- Kann freigegebene archivierte Datenpakete abrufen

Lebenszyklus von Datenpaketen



Endpunkte

- REST-basiertes API mit Antworten im JSON-Format
 - Dokumentation unter <https://www.radar-service.eu/de/radar-api>
 - Entwicklung: <https://dev-radar.fiz-karlsruhe.de/radar/api>
 - Produktion: <https://www.radar-service.eu/radar/api>
- Methoden für alle zentralen Entitäten
 - User
 - Contract
 - Workspace
 - Dataset
 - Folder
 - File

Fragen?

www.radar-service.eu
www.radar-projekt.org



Except where otherwise noted,
content on this site is licensed under
a Creative Commons Attribution 4.0
International License.