DATOS ABIERTOS: GUÍA DE USO DE LA API

Guía práctica institucionalizada para el acceso y explotación de datasets públicos — Versión 1.0 (Julio 2026)

1. Qué es BA Data y la API institucional

BA Data constituye el portal oficial de datos abiertos del Gobierno de la Ciudad de Buenos Aires (data.buenosaires.gob.ar). A través de esta plataforma, se centralizan, publican y distribuyen de manera gratuita conjuntos de datos estructurados ("datasets") generados por las diversas reparticiones gubernamentales, abarcando áreas críticas tales como movilidad, presupuesto, arbolado público, salud, seguridad vial y medio ambiente.

Si bien los archivos pueden descargarse de forma manual e independiente en formatos convencionales (CSV, Excel, JSON, Shapefile), el ecosistema dispone de una Interfaz de Programación de Aplicaciones (API). Esta herramienta automatiza las consultas y permite la extracción directa de información mediante peticiones estandarizadas basadas en URLs, devolviendo respuestas estructuradas en formato JSON optimizadas para el procesamiento computacional autónomo, prescindiendo de la navegación web tradicional.

Arquitectura Tecnológica: BA Data funciona sobre CKAN (Comprehensive Knowledge Archive Network), el software de código abierto más usado en el mundo para portales de datos gubernamentales. La infraestructura actual opera bajo la versión de producción CKAN 2.11.0, lo que garantiza un estándar bien documentado y con una sólida comunidad global.

Beneficios de la Consulta Automatizada por API:

- **Automatización integral:** Permite estructurar reportes o tableros de control que se actualizan de forma autónoma tras cada publicación oficial de datos nuevos.
- **Reproducibilidad científica:** Un investigador puede documentar con precisión matemática los parámetros de consulta empleados, asegurando la replicabilidad del análisis.
- **Eficiencia de escala:** Mitigación del consumo de ancho de banda e infraestructura al interactuar directamente con los recursos requeridos, evitando descargas innecesarias de archivos enteros de gran volumen.
- **Integración nativa:** Conexión directa del repositorio público con canalizaciones de datos (pipelines), dashboards organizacionales o cuadernos de código.

2. Guía de navegación por perfil de usuario

Para optimizar el uso de esta guía de acuerdo a sus competencias técnicas y objetivos, localice su perfil operativo:

- **Vecinos y Ciudadanos:** Orientado a visualizar o descargar datos específicos de su barrio sin requerir destrezas de programación. (Secciones 3 y 8).
- **Periodistas:** Búsqueda de datos confiables, perfectamente citables y aptos para una rápida diagramación visual. (Secciones 3, 6 y 8).
- **Académicos e Investigadores:** Enfoque centrado en la consistencia metodológica, análisis de series históricas y reproducibilidad científica. (Secciones 4, 6 y 7).
- **Analistas de Datos:** Integración fluida de la API en pipelines, cuadernos analíticos (Notebooks) o planillas automatizadas. (Secciones 4, 5 y 7).

• **Desarrolladores de Software:** Construcción de aplicaciones, plataformas independientes o servicios complejos sobre la API cruda. (Secciones 5 y 7).

• **Agentes de la Administración Pública:** Personal interno del GCBA responsable de estructurar, actualizar y mantener datasets institucionales. (Secciones 4 y 7).

3. Explotación de la API sin requerimientos de programación

No se requiere experiencia en desarrollo de software para extraer valor de la API institucional. Dispone de las siguientes rutas alternativas de acceso:

• **Opción A (Consulta vía Navegador):** Cualquier URL de la API puede ejecutarse de manera directa en un navegador web convencional. El sistema devolverá de inmediato la respuesta cruda estructurada en formato JSON, útil para inspecciones iniciales rápidas.

• **Opción B (Integración en Google Sheets):** Las hojas de cálculo de Google pueden capturar de manera directa archivos de tipo CSV publicados en BA Data mediante el uso de la función nativa: =IMPORTDATA("URL_DE_DESCARGA_DEL_RECURSO").

• **Opción C (Excel con Power Query):** Implemente la ruta *Datos* → *Obtener datos* → *Desde archivo* → *Desde CSV* (o seleccionando *Desde web* e insertando la URL de descarga). Power Query estructurará las columnas de manera automatizada y permitirá programar actualizaciones cíclicas cada vez que se abra la planilla.

• **Opción D (Descarga Convencional):** La vía directa habitual consiste en acceder a la interfaz del portal web institucional, seleccionar el formato de preferencia (CSV, XLSX, JSON, SHP) y ejecutar la descarga del archivo estático completo.

4. Conceptos estructurales de la API CKAN

4.1 La "Action API"

La API de CKAN opera bajo una arquitectura uniforme y parametrizada denominada Action API, donde cada acción funcional (listar, buscar, mostrar) equivale a una función específica a la cual se le suministran parámetros clave-valor a través de la URL de petición:

```
https://data.buenosaires.gob.ar/api/3/action/<nombre_de_la_accion>?parametro=valor
```

4.2 Formato Uniforme de Respuesta

Todas las transferencias de salida se codifican estrictamente en formato JSON, adoptando el siguiente esquema general estandarizado:

```
{ "help": "https://data.buenosaires.gob.ar/api/3/action/help_show?name=package_show",
"success": true, "result": {...} }
```

Nota de control: Frente a fallos operativos (parámetros erróneos o recursos inexistentes), la clave "success" computará como false y se anexará un objeto detallado bajo el nodo "error". Se aconseja validar sistemáticamente este estado previo al parsing de los resultados.

4.3 Políticas de Autenticación y Acceso

La totalidad de los endpoints orientados a la consulta y extracción de información (como package_show o package_search) son estrictamente de carácter público y **no requieren tokens de autenticación ni API Keys**. El uso de credenciales queda reservado exclusivamente a las acciones de escritura (creación o modificación de datasets), restringidas a los organismos publicadores oficiales del GCBA.

4.4 Clasificación Operativa de Componentes

Es crucial diferenciar la naturaleza de la información que se está consultando para optimizar los flujos de trabajo técnico:

Concepto	Definición Técnica	Endpoint / Acceso
Metadatos	Información estructural del dataset: título, descripción, repartición responsable, licenciamiento vigente y catálogo indexado de recursos con sus respectivas URLs de descarga.	package_show
Datos (Archivos Estáticos)	El contenido tabular o georreferenciado completo y unificado, empaquetado en archivos tradicionales (CSV, XLSX, SHP, GeoJSON) estructurados para su descarga total.	URL directa de descarga del recurso

4.5 Paginación y Restricciones de Volumen

Las respuestas asociadas a listados extensos u operaciones de indexación masiva implementan de forma nativa esquemas de paginación mediante el uso controlado de los parámetros de consulta corporativos rows (especifica el límite o volumen de registros a transferir por lote) y start (indica el índice de desplazamiento o posición inicial de la lectura).

5. Referencia técnica y especificación de endpoints

A continuación, se detallan los endpoints principales de consulta mediante ejemplos prácticos basados en el dataset real de *Arbolado Público Lineal* de la Dirección General de Arbolado del GCBA (identificador unificado: arbolado-publico-lineal).

5.1 Catálogo General de Datasets (package_list)

Recupere el listado completo de identificadores únicos correspondientes a los conjuntos de datos publicados en el portal institucional.

```
Petición (GET): https://data.buenosaires.gob.ar/api/3/action/package_list
```

5.2 Búsqueda Indexada y Filtrado Avanzado (package_search)

Permite ejecutar búsquedas complejas por coincidencia de texto e incorporar filtros de estructura avanzada.

```
Petición (GET): https://data.buenosaires.gob.ar/api/3/action/package_search?q=arbolado&rows;=5
```

Parámetros de optimización: Use q para búsquedas de texto libre y fq para restricciones condicionales exactas sobre campos específicos (ejemplo: fq=organization:direccion-general-arbolado).

5.3 Consulta Detallada de Metadatos y Recursos (package_show)

Constituye el endpoint de mayor relevancia operativa. Provee de forma integrada el título, estado de vigencia, licencias institucionales y las URLs definitivas de descarga de los archivos de datos.

```
Petición (GET):
https://data.buenosaires.gob.ar/api/3/action/package_show?id=arbolado-publico-lineal
```

5.4 Endpoints Complementarios de Consulta

• organization_list: Devuelve el catálogo unificado de las dependencias gubernamentales del GCBA que actúan como unidades de publicación oficial.

• organization_show?id=...: Filtra y expone de forma exclusiva la totalidad de los datasets asociados a una organización específica.

• group_list: Mapea las clasificaciones y taxonomías temáticas del portal (ejemplo: movilidad, ambiente, salud).

• resource_show?id=...: Inspecciona las propiedades y metadatos individuales de un único recurso sin necesidad de solicitar el dataset contenedor completo.

• tag_list: Retorna el inventario de etiquetas normalizadas empleadas para la indexación cruzada de datos.

5.5 Integración Algorítmica de Referencia en Python

```
import requests
import pandas as pd

base_url = "https://data.buenosaires.gob.ar/api/3/action/"

# 1. Ejecución de la petición y extracción del nodo de metadatos
response = requests.get(f"{base_url}package_show", params={"id": "arbolado-publico-lineal"})
dataset_meta = response.json()["result"]

# 2. Localización dinámica de la URL del recurso estructurado en formato CSV
csv_url = next(res["url"] for res in dataset_meta["resources"] if res["format"] == "CSV")

# 3. Consumo directo de la estructura tabular mediante la librería Pandas
df = pd.read_csv(csv_url)
print(df.head())
```

5.6 Integración Algorítmica de Referencia en R

```
library(httr)
library(jsonlite)

base_url <- "https://data.buenosaires.gob.ar/api/3/action/"

# 1. Petición HTTP GET estructurada
response <- GET(paste0(base_url, "package_show"), query = list(id = "arbolado-publico-lineal")
)
dataset_meta <- content(response, as = "parsed")$result

# 2. Aislamiento e indexación de la URL del recurso CSV
csv_url <- Filter(function(res) res$format == "CSV", dataset_meta$resources)[[1]]$url

# 3. Ingesta estructurada y lectura tabular
df <- read.csv(csv_url)
head(df)
```

6. Matriz de herramientas recomendadas por perfil profesional

Perfil Técnico	Herramientas Preferenciales	Capacidades Operativas e Integración
Ciudadanos / Periodistas	Google Sheets, Microsoft Excel, Datawrapper, Flourish	Explotación ágil libre de entornos de programación, modelado rápido de visualizaciones y facilidad de compartición.
Periodistas de Datos	Google Sheets, Datawrapper, OpenRefine	Punto óptimo de equilibrio entre velocidad de procesamiento y prolijidad metodológica en la depuración de datos anómalos.
Investigadores / Analistas	R (tidyverse), Python (pandas), entornos Jupyter / RMarkdown, Power BI, Looker Studio	Garantía de reproducibilidad científica: el código documenta de forma matemática la extracción y el análisis. Conexión nativa con dashboards de actualización automatizada.
Desarrolladores	Librería oficial ckanapi (Python), Postman, Insomnia, entornos de caché HTTP	Abstracción nativa de la Action API mediante funciones nativas del lenguaje, optimizando la integración en arquitecturas de gran escala.
Analistas GIS / Georreferenciados	QGIS (Open Source), ArcGIS	Soporte nativo y procesamiento avanzado de capas vectoriales complejas (Shapefiles, GeoJSON) publicadas de forma recurrente por el GCBA.

Sugerencia de integración: Para entornos corporativos que requieran potencia analítica intermedia sin codificación, la plataforma Looker Studio puede conectarse directamente a una hoja de Google Sheets alimentada mediante la función IMPORTDATA, consolidando un dashboard interactivo de actualización cíclica.

7. Código de buenas prácticas y optimización de consultas

• **Atribución Institucional y Rigor en Licenciamiento:** Es de carácter obligatorio citar la fuente formal y el tipo de licencia provisto bajo la clave license_title (habitualmente *Creative Commons Attribution*). Para toda publicación académica o de prensa, consigne de forma explícita el nombre del dataset, el organismo de origen responsable y la fecha exacta del proceso de descarga.

• **Control de Versiones y Trazabilidad Temporal:** Dado que los catálogos públicos están sujetos a actualizaciones dinámicas, registre el valor del campo metadata_modified provisto por el endpoint package_show; almacene este parámetro junto con sus salidas analíticas para salvaguardar la replicabilidad de sus modelos.

• **Políticas de Mitigación de Carga en Servidores:** Evite la ejecución masiva de peticiones o descargas recurrentes en paralelo. Introduzca pausas e intervalos temporales controlados entre solicitudes analíticas secuenciales e implemente sistemas de almacenamiento en caché local para recursos estáticos previamente descargados.

• **Uso Imperativo de Protocolos Seguros:** Utilice sistemáticamente el cifrado seguro HTTPS para interactuar de forma íntegra con la totalidad de las URLs y endpoints de la API corporativa.

• **Consistencia en Datos Geoespaciales:** Previo al cruzamiento o superposición de capas georreferenciadas, valide con precisión matemática el Sistema de Referencia de Coordenadas (CRS) nativo del archivo Shapefile o GeoJSON bajo análisis.

• **Flexibilidad ante Variaciones de Esquema:** No presuponga una inmutabilidad absoluta en la arquitectura o disposición estructural de las columnas entre versiones temporales distintas de un mismo conjunto de datos (ejemplo: arbolado histórico frente a relevamientos recientes). Valide de forma programática las cabeceras de los archivos antes de procesar scripts automatizados heredados.

8. Tutorial de implementación práctica paso a paso

Desarrollo metodológico práctico orientado a la estructuración de un gráfico analítico simple, tomando como base el dataset real de *Arbolado Público Lineal* de la Ciudad Autónoma de Buenos Aires.

Ruta A: Implementación Analítica sin Código (Google Sheets)

1. **Localización del dataset:** Acceda a la plataforma centralizada data.buenosaires.gob.ar e indexe el término "arbolado público lineal".

2. **Extracción del enlace de descarga:** Localice el recurso estructurado en formato CSV, haga clic derecho sobre el botón interactivo "Descargar" y seleccione la opción "Copiar dirección del enlace".

3. **Inyección automatizada de datos:** En una celda de destino vacía dentro de su planilla de cálculo, declare la función analítica: =IMPORTDATA("URL_COPIADA").

4. **Modelado de datos unificado:** Seleccione el rango de datos importado y ejecute la ruta *Insertar* → *Tabla dinámica*. Configure las filas agrupadas bajo la columna taxonómica de la especie y asigne al campo de valores un conteo volumétrico de registros por ejemplar.

5. **Consolidación gráfica:** Ejecute *Insertar* → *Gráfico* tomando como origen la matriz dinámica construida para obtener de inmediato el ranking oficializado de las especies arbóreas predominantes en el territorio de la Ciudad.

Ruta B: Implementación Algorítmica con Python

Para desplegar el pipeline analítico automatizado de extracción y representación visual, ejecute el siguiente bloque de sentencias en su entorno:

```
import requests
import pandas as pd
import matplotlib.pyplot as plt

base_url = "https://data.buenosaires.gob.ar/api/3/action/"

# 1. Recuperación controlada de metadatos estructurales
meta_data = requests.get(f"{base_url}package_show", params={"id": "arbolado-publico-lineal"}).
json()["result"]

# 2. Filtrado dinámico para la captura de la URL de descarga del recurso CSV
csv_url = next(r["url"] for r in meta_data["resources"] if r["format"] == "CSV")

# 3. Descarga e ingesta automatizada en estructura Dataframe de Pandas
df = pd.read_csv(csv_url)

# 4. Procesamiento estadístico: extracción de las 10 especies predominantes
top_especies = df["nombre_cientifico"].value_counts().head(10)

# 5. Renderizado y exportación de la gráfica de barras horizontales institucional
top_especies.plot(kind="barh", title="Especies de árboles más comunes en CABA")
plt.tight_layout()
plt.show()
```

9. Repositorios, soporte técnico y canales de contacto

Recurso Institucional	Canal Técnico / Enlace de Acceso
Portal Oficial de Datos Abiertos	data.buenosaires.gob.ar
Punto de Entrada de la Action API	data.buenosaires.gob.ar/api/3/action/
Documentación Oficial del Framework	docs.ckan.org
Repositorio de Código del GCBA	github.com/datosgcba
Canal de Contacto Técnico Directo	datosabiertos@buenosaires.gob.ar

Historial de revisiones técnicas del documento:

Fecha de Publicación	Versión	Descripción Técnica de Modificaciones
02/07/2026	1.0	Emisión unificada de la documentación técnica y guía de explotación pública de la API de BA Data (optimizada sin componentes analíticos de consultas de almacenamiento persistente / datastore).